Hochschule für Angewandte Wissenschaften Hamburg
Hamburg University of Applied Sciences

Master Thesis

Can Ulaş

Conceptional design of a decentralised control environment for track based automated guided vehicles

Fakultät Technik und Informatik
Department Maschinenbau und Produktion

Faculty of Engineering and Computer Science
Department of Mechanical Engineering and Production Management

Can Ulaş

Conceptional design of a decentralised control environment for track based automated guided vehicles

Masterarbeit eingereicht im Rahmen der Masterprüfung

im Studiengang Produktionstechnik und -management
am Department Maschinenbau und Produktion
der Fakultät Technik und Informatik
der Hochschule für Angewandte Wissenschaften Hamburg

Erstprüfer: Prof. Dr.-Ing. Hans-Joachim Schelberg

Zweitprüfer : Prof. Dr.-Ing. Henner Gärtner

Abgabedatum: 23.10.2018

Zusammenfassung

Can Ulaş

Thema der Masterthesis

Konzeptionelle Gestaltung einer dezentralen Steuerungsumgebung für spurgebundene fahrerlose Transportsysteme

Stichworte

Blockchain, Fahrerlose Transportsysteme, Dezentralität, Industrie 4.0

Kurzzusammenfassung

Der Themenbereich der Blockchain im Zusammenhang mit industriellen Anwendungen ist in den letzten Jahren immer stärker in den Fokus der Produktion gerückt. Die Blockchain-Technologie beschäftigt sich mit der Dezentralisierung von Daten in einer nicht vertrauensvollen Umgebung. Ein weiteres Alleinstellungsmerkmal ist die Unveränderbarkeit der Daten, sobald sie einmal veröffentlicht wurden. In dieser Arbeit werden zunächst die Grundlagen der Blockchain sowie Grundlagen der Dezentralisierung von Produktionsprozessen und fahrerlosen Transportsysteme beleuchtet. Anhand eines ausgewählten Beispiels werden die Machbarkeit und der Grad der Implementierung eines solchen Umfelds, mittels einer industriellen Anwendung vorgestellt. Ziel ist es zu überprüfen, ob aus der Kombination der Technologien ein Mehrwert für die Industrie erzielbar ist. Zu diesem Zweck wird ein Konzept entwickelt und für diese Anwendung untersucht.

Can Ulaş

Title of the paper

Conceptional design of a decentralised control environment for track based automated guided vehicles

Keywords

Blockchain, Automated guided vehicles, decentralisation, Industry 4.0

Abstract

In recent years, blockchain technology in connection with industrial applications has increasingly moved into the focus of production. Blockchain technology deals with the decentralisation of data in a non-trusted environment. Another unique selling point is the unchangeability of the data, once it has been published. In this thesis, the fundamentals of blockchain as well as the fundamentals of decentralisation of production processes and automated guided vehicle systems are examined. The feasibility and implementation of such an environment is presented by means of an industrial application using a selected example. The aim is to check, whether the combination of these technologies can generate added value for industry. For this purpose, a concept will be developed and investigated upon the discussed application.

Contents

List of Figures

List of Abbreviations

AGV	Automated Guided Vehicle
BPIIoT	Blockchain Platform for Industrial Internet of Things
CPPS	Cyber-Physical Production System
CPS	Cyber-Physical System
CPU	Central Processing Unit
CTA	Container Terminal Altenwerder
DAO	Decentralised Autonomous Organisation
DGPS	Data by Means of the Reference Signal
EIP	Ethereum Improvment Proposa
I FFG	Friendly Finality Gadget
GPS	Global Positioning System
IIoT	Industrial Internet of Things
IoT	Internet of Things
M2M	Machine-to-machine
PDGPS	Precision Data by Means of the Reference Signal
PoS	Proof of Stake
PoW	Proof of Work
RFID	Radio-Frequency Identification
TCP/IP	Transmission Control Protocol/Internet Protocol

1. Executive summary

This research thesis provides an analysis and evaluation of the implementation and sense of purpose of blockchain technology in industrial environments for the purpose of material tracing with the use of automated guided vehicles. For this investigation a blockchain environment is proposed in reference to an industrial environment. By using an example, a custom crypto currency is deployed theoretically. Other methods for this work include the consideration of certification of sensitive and confidential products by using blockchain technology. Results of the analysis show that the use cases of blockchain technology are poorly given in general. In particular, the only working sector seems to be using blockchain technology for financial purposes, such as asset management or payment processes in form of crypto currencies.
The thesis finds the prospects of blockchain technology in industrial fields not positive. The lack of benefit compared to current working technologies is too big and requires further technological investigation. Recommendations discussed include:

- Implement the approach of decentralisation approached
- Using current technologies for first applications
- Application of RFID tags for tracking purposes

The thesis also investigates the fact that the analysis conducted has limitations for real world application. Some of the limitations include:

- Inappropriate business line
- Top-down company hierarchy
- Missing reference values for a successful application

2. Introduction

In the course of the fourth industrial revolution, the topic of decentralisation has received huge attention. Decentralised aspects are often found in manufacturing companies, where the production process is not controlled by a central authority. Instead, communication takes place between individual entities. This works on the basis of if-then results. A relatively unexplored technology here is the blockchain technology. It gained publicity through crypto currencies. A special feature of the blockchain is that information is invariably stored in a data chain. Since each block refers to the previous block, the change of a block leads to a holistic change. This principle is transferred to a logistical model in the context of this work. With the help of the blockchain, an environment is created in which transported goods can be reliably tracked and thus losses in the supply chain can be avoided. For this purpose, a new project, the *HAW Coin*, will be suspended on the basis of an existing blockchain. This will enable the tracking of transported goods within the blockchain.

This work is divided into six chapters. The chronology of chapters essentially reflects the actual sequence of execution. Following the introduction, the basic principles that are important for this work, are presented in chapter 3. initially, the structure and functionality of the blockchain are discussed in section 3.1. Sections 3.2, 3.3 and 3.4 are dedicated to refreshing the technical principles common in decentralised production environments.

In chapter 4, an intelligent environment is introduced and the process is presented on the basis of existing prototypical examples. Section 4.1 portrays the motivation behind this work. Ensuing in section 4.2 deals with the fusion of blockchain technology and industrial applications. In section 4.3 the implementation of the blockchain in this environment is performed and checked theoretically. In section 4.4 the scenario is extended to a multiple vehicle environment.

Subsequently, a further field of application is presented in section 4.5.

A summary of the results can be found in chapter 5. The critical appraisal is given in chapter 6. Here the results are examined and evaluated from a critical perspective. In the last chapter 7, a recommendation for possible further developments is given.

3. State of the art

Within this elaboration, the applicability and implementability of the blockchain technology shall be investigated within the framework of a decentralised logistical system. Therefore an introduction into the the state of the art is mandatory in order to fully understand the concept. First in section 3.1 the blockchain fundamentals are imparted including the extra functionalities that can be implemented into blockchain technology. Since the technology of the blockchain is relatively new to the general public it is executed more extensively. Followed by the technical basics in section 3.2, regarding to the automated guided vehicles that play a keyrole in this contemplation. Within the same discussion, key information are alluded in the framework of *Industry 4.0* and decentralised production processes in section 3.3 and section 3.4.

3.1. Blockchain

Over the past few years the blockchain got more and more attention. *Bitcoin* first put the blockchain into a broad public by releasing the whitepaper for a blockchain based crypto currency. Since then, a vast amount of other coins and tokens were released. In the following sections the blockchain is extensively explained which is needed in order to follow the content of the continuing part.

3.1.1. Fundamental knowledge

The tern *blockchain* results in the architecture of the technology. Having blocks chained against to each other creating a chain. The blocks themselves can consist of different information. In case of crypto currencies, among other things the transactional information is stored in the block. As seen in figure 3.1 the information of each block is dependant of the previous block. Hence, a change of information of a block that is already released into the blockchain is not possible without producing an error in the chain. This makes the blockchain an irreversible technology. Once a transaction is made, there is no way to undo it unless one entity controls at least 51% of the network power. Every block in the blockchain ecosystem

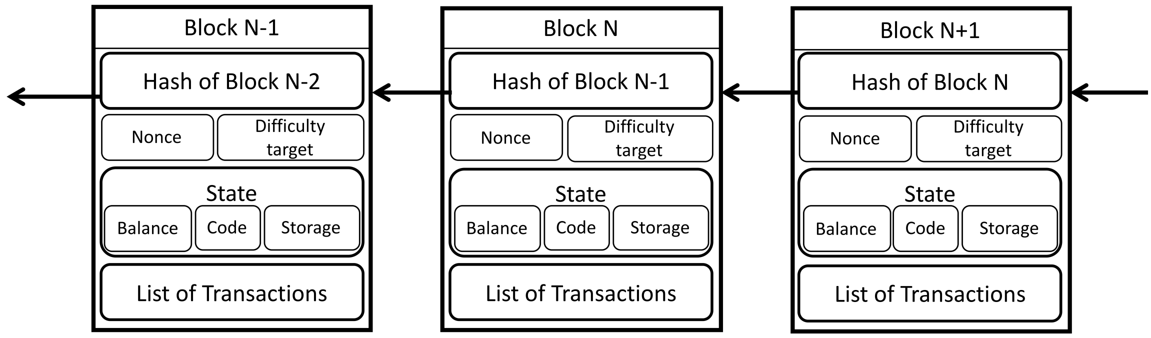


Figure 3.1.: Simplified illustration of blockchain segments [Dre17]

needs to be changed, if a block shall be changed afterwards. Therefore the blockchain gets more and more secure and resilient to attacks, as it grows in amount of valid blocks. [Dre17]

3.1.2. Blockchain variations

As blockchains represent a new ecosystem and also a complete new industry, the creators of the system can decide on their own, how the blockchain is going to be constructed. There are four basic blockchain variations that can be distinguished from. Figure 3.2 shows the four different scenarios that can be played out. The distinction is made between public/private or permissioned/permissionless. In the following, all four combinations are explained into further detail.

	Reading Access and Creation of Transactions	
Writing Access	**Everyone**	**Restricted**
Everyone	Public & Permissionless	Private & Permissionless
Restricted	Public & Permissioned	Private & Permissioned

Figure 3.2.: Overview of possible blockchain variations [Dre17]

Public permissionless blockchain

The public permissionless blockchain approach is the best proven and most widely used variant to date. *Bitcoin* was the first blockchain project of this kind and has engendered several clones and offshoots which basically all use a native token, in the sense of a crypto currency for exchange and payment transactions. As an important feature it should be mentioned that this form of blockchain usually uses the so-called *Proof of work* (see section 3.1.7) as a consensus mechanism. This includes the calculation of a certain hash value for block generation is the goal and newly created tokens are issued in the form of crypto currency as a reward for the resources used. People or organisations who contribute to the system with new tokens are called miners. A possible alternative mechanism to consensus building (see section 3.1.4) is the process of *Proof of stake* (see section 3.1.8) [Dre17].

Public permissioned

With the public permissioned-based blockchain approach, the complete management of the blockchain, as well as the validation of the transactions, is handled by a trustworthy group. This group can vary in its composition again and again. However, the process of identifying a trustworthy actor to take on these tasks must be clearly defined and supported by all participants. One of the many alternatives to *Proof of work* can also be used as a consensus mechanism, as the identities of all network participants are also known here. This form of blockchain can also be described as *consortium blockchain*, even if in this case the reading right is declared as public and thus a broad user population is permitted. [Dre17]

Private permissioned

The basic idea behind this variant of a blockchain is not to give everyone the possibility to view the database .Only a certain group of participants is allowed for transaction execution or validation as well as for block creation. Reading rights can be granted separately to other entities, regulators or auditors. Such a construct is often referred to as the *blockchain consortium*. Due to the fact that the identities of all actors are known, this enables the use of alternative consensus mechanisms that are less resource-intensive than the *Proof of work*, which are not based on competition and thus enable faster transaction processing. It is also important to note that in license-based systems such as this, no use of native tokens is required, as the validating and block generating nodes can be remunerated on an extended basis. [Dre17]

Private permissionless

Participants can neither benefit from open accessibility nor from additional options of a closed system, such as a rollback [Zep16]. This type of blockchain makes little sense from a logical point of view and seems to have no real purpose and value, which is why it is not considered further. Finally, the degree of centralisation of the respective blockchain variant can be deduced from these presented differentiation options, as shown in figure 3.2. It can be seen that the more restrictions a blockchain contains, the more it deviates from its core principles, i.e. decentralisation as well as administration and trustless functioning. [Dre17]

3.1.3. Double spending problem

Nowadays, digital goods and products are often very easy to duplicate and have no differences to the original product. This may be an advantage for emails, photos and self-made movies, as they can be shared so quickly with friends and acquaintances. However, with digital goods that need to be protected, as this can lead to major problems. One example is the money that would suddenly become available to an infinite extent in the event of arbitrary duplication and expenditure. On the one hand this would be illegal and criminal consequences would have to be expected, on the other hand, the money would loose value very quickly due to the loss of confidence and the infinite amount availability. This is known as the problem of the double edition. [Dre17]

In central systems, an authority as intermediary always controls the system and prevents such incidents. Because in general, only what is actually available to you should be allowed to be used or passed on. In distributed networks, however, this protection mechanism no longer exists due to the lack of such a central authority. Therefore a different solution must be found for the decentralised system of the blockchain to guarantee integrity and trust within the network [Roß16]. One possibility is the use of a consensus. This topic is elaborated in the following section.

3.1.4. Distributed consensus

A distributed consensus is a mechanism, that prescribes how mutually distrusting and anonymous network nodes agree on the validity of recently executed transactions in terms of their chronological order [Zep16]. In the event of a conflict between two propagated transactions with the same unit, the transaction that moved the corresponding unit first is considered valid [Nak08]. Furthermore, valid transactions should have certain characteristics [Gar15].

- Consistency: Transactions must correspond to the current state of the system. If someone owns 1000 units of a digital currency or another digital good he cannot send more than 1000 units. [Dre17]
- Authorisation: Transactions must be approved. This means that it is clearly defined who may carry out transactions and how one has to alloy oneself for it. [Dre17]
- Unchangeability: Transactions cannot be modified. Once a transaction has been saved in the blockchain, it is impossible to change it afterwards. [Dre17]
- Finality: Transactions are irreversible. Once stored in the blockchain, they cannot be undone or deleted. [Dre17]
- Censorship Resistance: If a transaction matches the protocol, it must be included in the blockchain irrelevant of its purpose or value received. [Dre17]

Another scenario in which the distributed consensus can be used would be a division of the blockchain. In such a case the blockchain splits into two branches and the network nodes would have to decide by consensus which of the two branches will be maintained. For a successful division of the blockchain, the network needs at least 51% approval [Zep16]. This phenomenon is also called a *fork*. Current known forks are *Bitcoin Cash* as a fork of *Bitcoin* and *Ethereum Classic* as a fork of *Ethereum*.

3.1.5. Cryptography

The blockchain is primarily based on two fundamental elements of cryptography. Asymmetric encryption or digital signatures and hash functions [Sch16]. Asymmetric encryption is the public-key encryption method developed by [DH76] in the 1970s that enables secure key exchange via open, unsafe communication channels [Mün09]. A random, corresponding key pair is created by means of a certain encryption algorithm, which consists of a public and a secret, private key [Fra14]. Such a key pair can be used to create a digital signature [Sch16]. *Bitcoin* and other crypto currencies use this encryption method to authorise transactions. Each participant of the network has several such key pairs. The public key, which must always be specified by the transmitter at the beginning of each transaction and is identical to the account number of a bank account, can basically be viewed by any other participant in the network. However, it is not possible to draw any conclusions about the private key. To release the transaction, it must be signed with the corresponding private key [Swa15]. This should always be kept very carefully, as it allows access to the monetary values and data stored in the public key. To validate a signature, it is sufficient to know the sender's public key [Gar15].

Hash functions are algorithms that convert a character string of any length into a character string of a certain length, which is called a hash value. Hash functions are deterministic, which means that the same input data always give the same hash value. However, hash functions are also one-directional which means that it is impossible to conclude the input by using the hash function. A minimal change of the input data, causes a strongly changed hash value [Sch16][Dre17]. Thus, each hash value generated is unique and resembles a digital fingerprint. Since the blockchain is a database in which new data entries or transactions are made block by block, this element of cryptography is used to protect the blocks and thus also any transactions from subsequent manipulation. The principle is to generate a hash value for each block as seen in figure 3.1. This sequential linking of the blocks to each other makes any manipulation very time-consuming, since each hash value of a subsequent block depends on its previous block.
If an attempt is made to manipulate a block, this would lead to a change in the hash value, whereupon all other subsequent blocks would no longer be consistent and would also have to be manipulated. This should also happen on the majority of all nodes in the network, and also before the blockchain is updated again by another block. Another additional hurdle is the effort to generate a block depending on the consensus mechanism, which is associated with a lot of computing power in the case of the *Bitcoin* blockchain for example. It can therefore be concluded that the blockchain is protected against any form of external and internal manipulation [Roß16].

3.1.6. Trustless control environment

Unfortunately, it is not always possible in a distributed peer-to-peer system to follow the principle of considering the transaction as valid, which in the case of a double output was first propagated with the corresponding values in the network. This is because the distribution of nodes due to the network structure means that transactions can only gradually penetrate the network. This means that it is possible that a second propagated transaction may arrive first at some nodes and thus be considered valid by them. This is a well-known problem of computer science, the so-called "Byzantini general problem" [LSP82], which made it impossible before the invention of *Bitcoin* and its underlying blockchain technology to achieve unity between participants in distributed network systems [Dre17]. The solution to this problem by using the blockchain can be achieved with consensus mechanisms. However, the decision is not made for each individual transaction, but for the blocks in which the transactions are contained. Newly released transactions are first buffered in a transaction pool, where they wait to be added by one miner to the next block and added to the existing chain of blocks by consensus decision [Roß16]. Thus, con-sensitivity mechanisms describe the way in which the participants in a distributed blockchain network decide which blocks are to be used to extend the current state of the chain. The consensus mechanisms may differ in their approaches.

3.1.7. Proof of work

With the *Bitcoin* protocol, which made blockchain technology possible and popular in the first place, the *Proof of work* mechanism was developed, which is still the most frequently used method of consensus building for blockchains. The nodes of a network compete with each other in order to bundle the latest transactions and add them to the database in the form of blocks. This competition is about finding a suitable solution to a mathematical puzzle. Randomly selected character strings (nonce) are iterated until the hash value of the block corresponds to the network targets. *Satoshi Nakamoto* described the process in his working paper published in 2008 as follows:

1. New transactions are forwarded to all nodes [Nak08].
2. Each node bundles these new transactions in a block [Nak08].
3. Each node works on a *Proof of work* puzzle for its block [Nak08].
4. When one node has solved the puzzle, the block is sent to all other nodes [Nak08].
5. Nodes accept the block only if all contained transactions are valid, which means that the transactions must have the properties described in section 3.1.4 [Nak08].

6. Nodes accept the block by building the following block in the chain on the hash of the accepted block [Nak08].

A solution to this mathematical puzzle can be very costly due to the massive use of special hardware, which consumes a corresponding amount of energy. However, checking the solution by the other nodes is quite simple [Gar15].

Since nodes, as described above, also try to create blocks in addition to validating transactions, they are referred to as miners. This is an allusion to gold mining, where gold is sought in mines with the help of appropriate equipment shafts in order to generate profit. Mining a block for the block chain is similar. Once a block has been successfully generated by solving the mathematical puzzle, the corresponding node receives a reward in the form of newly created units of crypto currency (tokens). On the one hand, to compensate for the costs incurred for hardware and energy consumption and on the other hand to serve as an incentive for further block generation. Furthermore, the block reward serves as protection against manipulation attempts within the system. Because a node that has more than 51% of the computing power of the entire network could try to change transactions in its favour. Howerver this would lead to a loss of confidence in the crypto currency with a corresponding price loss very probable [Swa15] [Dre17].

The node (miner) would thus endanger the assets it has generated so far. Therefore it is always advisable to follow the rules of the network, such as creating blocks with which the blockchain is updated and receive the reward. This type of consensus ensures that the nodes do stay honest, as manipulated blocks will not get a reward. This way the energy would have been used up for nothing [Nak08].

If two or even three nodes have found the solution to the puzzle at the same time, the blockchain is split for a short moment. In such a conflict of multiple blockchains, the generated blocks differ in the recipient address of the block reward, as well as in the transactions in the blocks [Six17].

Thus, due to the delayed network penetration, the other nodes sometimes receive different blocks on which they try to create further new blocks. The other two blocks are also recognised and stored as a precaution. In such a specific case of division, one rule is that the longest blockchain always wins and is considered the correct one.

Two of the three blockchains will expire as soon as one turns out to be the longest. Those nodes that were active on the two shorter blockchains will change to the longest blockchain accordingly [Nak08]. To ensure that the blockchain is always extended by new blocks within a certain time interval defined by the network, the difficulty of creating the block is adjusted at regular intervals. The more resources the nodes use to generate new blocks, the more difficult it is for them. If the use of resources is reduced, the difficulty of generating a block decreases [Nak08]. The biggest disadvantage of the *Proof of work*, which is also often the subject of discussion, is its energy consumption [Gar15]. For this reason, more and more alternative concepts for reaching a consensus have been developed over time.

3.1.8. Proof of stake

One of these alternative concepts is the *Proof of stake* approach. This form of consensus has already been used in a wide variety of cryptographic currencies. Some of these are e.g. *Peercoin, Blackcoin, Bitshares* and *Nxt* [Gar15]. Here, the probability of generating a block does not depend on the computing power of the node, but on its value-based share in the network. The larger the ownership share of a crypto currency, the greater the probability of generating a block [Sha18]. The idea behind this approach is that users with significant ownership also have a significant interest in maintaining and securing the blockchain. Because if manipulated, they would be the ones who would lose the most [Zep16]. [But16] wrote in his blog that an attacker would first have to buy up large amounts of the available tokens in order to successfully attack the blockchain. Once this has happened, however, the currency would rapidly lose value, which takes the whole project ad absurdum. From a financial point of view, the attacker can only lose [But16]. Since the *Proof of stake* process does not require costly hash value calculations, it is not only cheaper than *Proof of work*, but also faster. The time required to generate a new block is thus much shorter for the *Proof of stake*, which in return means faster confirmation of the transactions [But16].

Another big difference of a pure *Proof of stake* approach is that with the beginning of the first block of the chain all units of a crypto currency already exist. Therefore, the incentive system in such a system is limited to the transaction fees per block credited to the block producer. In hybrid models where both *Proof of stake* and *Proof of work* are used, new units of crypto currency continue to be issued as block reward [Gar15].

The most controversial discussed point of criticism of the *Proof of stake* is the so called *nothing-at-stake problem*. If, by mistake or due to a significant conflict between the network participants, the blockchain should be divided, it would be very likely that the participants would now continue to work on both branches. Compared to *Proof of work*, where the use of both branches for block generation is either much more resource-intensive, or reduces the chances of creating a block if the available resources are divided between the two branches, no additional costs arise with *Proof of stake*. The chances of generating a block are also not diminished. A solution to the dispute would therefore be completely impossible and would allow attackers to launch a wide variety of attacks on the network. There are several proposed solutions to this problem, but so far none based on a decentralised approach for a pure *Proof of stake* mechanism [Gar15].

3.1.9. Smart Contracts

Smart Contract, is a concept that is older than the blockchain itself. Computer scientist and crytographer Nick Szabo published the idea for smart contracts in 1997 in a scientific paper entitled "Formalizing and Securing Relationships on Public Networks" [Sza18]. In it, he defines *Smart Contract*s as computer-controlled transaction log that enable contractual details to be mapped into machine code using a script language and executed automatically (see Figure 3.3) [Sza18]. *Smart Contracts* are a key element, as is a substantial extension of the blockchain, which are at least as important in meaning as the invention of the *Hypertext Markup Language* (HTML), with which information can be linked and freely accessible on the Internet. Anyone who therefore does not understand the concept of *Smart Contracts* cannot understand the potential of the blockchain either [Mou16].

In the context of the blockchain as a medium for implementation, *Smart Contracts* are

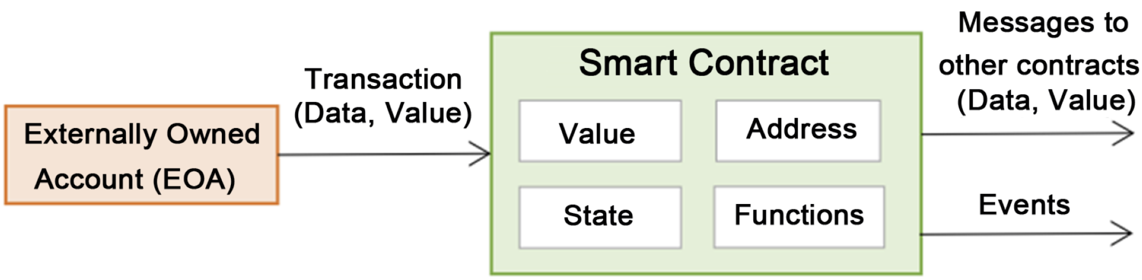


Figure 3.3.: Structure of a basic Smart Conract [BM16]

used as self-executing computer programs to store all clauses, agreements and obligations of a contract between two or more parties in the blockchain. Due to the transparency of blockchain networks, these codified contract contents can be viewed by everyone, can be

easily checked, monitored and carried out autonomously as soon as certain conditions have been met. The big advantage here is that no third party is required, which creates a contractual basis that significantly reduces costs and increases the degree of automation [Sch16]. The contracting parties do not necessarily have to be humans. [Swa15] distinguishes between two types of *Smart Contracts* is presented. On the one hand a human-to-human interaction and on the other hand the execution between technical units (machine-to-machine).
To make the mechanism and functioning of *Smart Contracts* more understandable, a description by [BM16] will be used. The two authors write that *Smart Contracts* are transactions containing tiny snippets of code. These intelligent contracts can be written in various higher programming languages. Language-specific compilers translate a written contract into byte code. Once translated, the contract is transferred to the blockchain, where it is assigned to a specially created address. Anyone who knows this address can now interact with the contract. *Smart Contracts* contain a series of executable functions and status variables.
These functions are executed as soon as someone sends a transaction to the address of the *Smart Contract* and it is included in a new block as confirmed. The transactions in return must contain parameters prescribed by the functions of the contract. When the function is executed, the state variables of the contract change depending on the logic implemented in the functions. Figure 3.3 shows how such a *Smart Contract* works [BM16].
The use and application possibilities of such intelligent contracts can be very diverse, as they not only transform existing business and administration processes in which data or information is archived, verified, authenticated, licensed or simply made available from the ground up, but also have the potential to enable completely new business models that were previously unfeasible or hardly imaginable.

3.2. Automated guided vehicles

Automated guided vehicles (AGVs) are already in broad use in a different industrial sectors. Especially for applications with a repetitive movement of materials over a certain distance is a typical use [MHI12].
In the following sections the current use of AGVs is explained briefly, followed by technical aspect in how AGVs work in section 3.2.2

(a) Light weight vehicle transporting carton boxes [Rät18]

(b) AGV handling cargo containers on the harbour of Hamburg/Altenwerder [Rät18]

(c) Heavy weight AGV transporting a car frame (Audi R8) [Bär14]

Figure 3.4.: different use cases for AGVs

3.2.1. Current use

Industrial use of AGVs is nothing special anymore. However, there is a high variety in the use cases. Automated guided vehicles (AGV) are already used in a broad scale in the industry and are deeply integrated in the supply chain of bigger companies, especially in the maritime sector [sha17]. The container terminal Altenwerder (CTA) in the port of Hamburg owns one of the most modern port infrastructure worldwide. 90 AGVs are operating on ground handeling the containers. Therefore over 19.000 transponders are embedded into the ground of the port area in order to prevent collisions between the vehicles [Plu18] [Lor18].
A great advantage of AGV use is not only the automation aspect, but also the fact that AGVs can operate day and night. This provides flexibility in the process as well [sha17]. This is one of many examples that shows, that using AGVs gives the operator a massive advantage.

3.2.2. Control technology

Depending on the environment, different control approaches can be pursued. The technology by which the vehicles operate is also to be considered for modularity and scalability. Some technologies are not as suitable to scale as other technologies.

Optical method

Optical control is met by installing reference objects, mostly in the form of a black line on the floor. A camera attached to the vehicle distinguishes the colour of the line it shall follow from the colour of the ground. Obviously the ground has to have another contrast in colour

in order for the camera being able to distinguish.
Optical processes achieve high driving accuracy through permanent course correction. The disadvantage is a susceptibility to contamination of the optical components and thus limited outdoor use. Scalability is given as for new tracks only a new line has to be installed on the floor.
The vehicles operate with batteries that have to be charged over time. In a scaled up environment the situation must ensure enough charging stations for all the vehicles. Also vehicle downtime can be an issue, when the number of operating vehicles does not match the demand in the production process.

Inductive method

This method of guiding driverless transport vehicles is particularly reliable due to the permanent course correction and cost-effective on the vehicle side due to the use of simple components. It is possible to realize the power supply of the vehicles on the road side, which eliminates the use of heavy accumulators. Furthermore, systems with stringline control (regarding changes of the track) are not flexible and they are very expensive on the track side [dub04].

Magnetic method

Another way of controlling the vehicle is to scan magnetic stripes or magnetic marks on the road surface. The change in the magnetic field of the current flowing through coils mounted in the vehicle is detected.

Control through GPS

In 1993, the U.S. Military established a worldwide available Satellite navigation system GPS (Global Positioning System), whose signals were encrypted and allowed an accuracy in the civil range of 100m. For civilian use, since May 2000, the signal is also available with non-artificial deteriorated position data. However, an accuracy of about 20m is not sufficient for a vehicle guidance. A higher accuracy of up to 5m can be achieved with the help of a second, fixed GPS receiver, which serves as a reference station. The neighbouring GPS receivers correct the received GPS position data by means of the Reference signal (DGPS) [Fau18]. A determination of the phase of the carrier signal of the individual satellites allows the accuracy to be increased to below 100 mm. This procedure is called Precision DGPS (PDGPS). Disadvantage of this technology is the high price. Furthermore, due to the shading in buildings and in confined spaces, trouble-free operation is not guaranteed [OBNMBDHF06].

Control through transponders

The control through transponders such as RFID is relatively new. An array of antennas measures the influence of the RFID tags and interpolates the data received. Through the

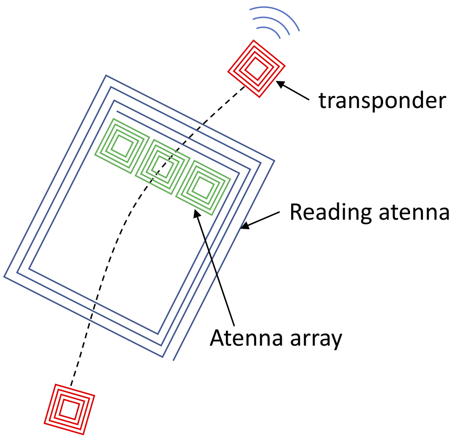


Figure 3.5.: Principle of control through transponders [OBNMBDHF06]

maximum of the signal strength the next position of the transponder can be assesed through interpolation [OBNMBDHF06]. To determine the route, the next transponder is then approached via a map that is available in the processing unit (CPU). This process is shown schematically in figure 3.5.

3.3. Industry 4.0

As the number in the term tells, *Industry 4.0* is the fourth revolution in modern manufacturing (see figure 3.6). It was first introduced publicly in 2011 under an initiative to enhance the german competitiveness in the manufacturing industry [Mar17].

It is important to distinguish it from the third industrial revolution, which puts the focus

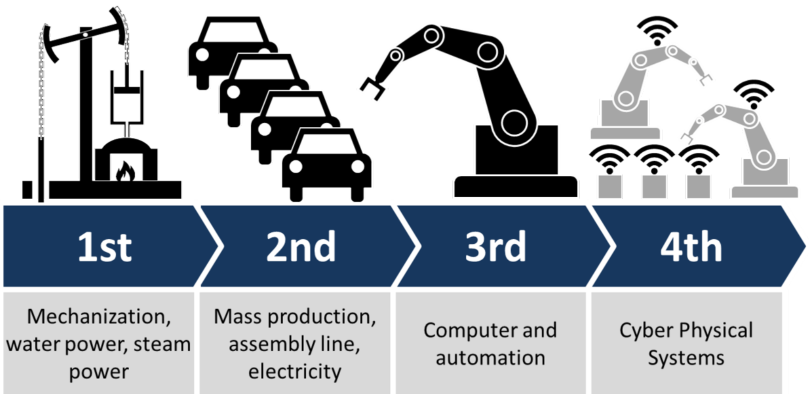


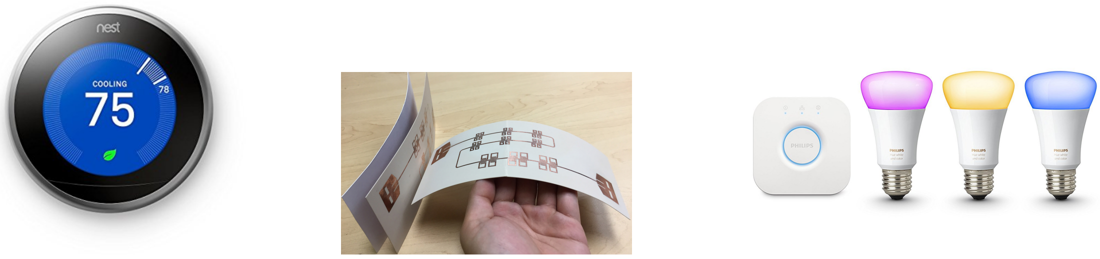
Figure 3.6.: Principle of control through transponders [Ros15]

on automation. The fourth revolution takes it one step further by setting the goal of letting machines work completely autonomously. This includes not only communication, but also maintenance, payment and more. The dependency of the human shall be as little as possible to allow a work shop to operate without any human. Decentralisation of manufacturing processes is also part of *Industry 4.0*. Nowadays centralised production sites are the standard, flexibility is not given in most cases. Decentralisation solves this problem by allowing to produce parts with a lot size equal to one.

3.3.1. Internet of Things

The term *Internet of Things* (IoT) describes the digitisation of products that do not have access to the digital world [JM15]. The origins lay in the use of *Radio Frequency Identification* (RFID). It was first used in the 1990's by Procter& Gamble. The majority of use cases were in the logistical department for tracking the goods along the supply chain via the internet [Rot16] [SSLL14].

Basically the *Internet of Things* describes the networking of simple devices to the internet. Besides mobile phones, which now exist almost only in form of smartphones and whole vehicles, various everyday life items like light bulbs [Lig18], cameras, doorbells and smoke alarms [Lab18b] are equipped with a logical unit which makes them smart and enables an internet access (see figure 3.7) [Rot16]. In relation to the industry the approach is the same. Components which were not "smart" yet get a smart component which connects them to



(a) Smoke detector by Nest inc. [Lab18b]

(b) Printable RFID tags by LiveTag [Lab18a]

(c) Smart light bulb by Phiilips Hue [Lig18]

Figure 3.7.: Example of smart everyday devices

the internet. This technology is being developed in high demand and has reached a level now where the cost of manufacturing reached a very low level. Xinyu Zhang a professor of electrical and computer engineering at the "UC San Diego Jacobs School of Engineering" [1] developed printable RFID tags (see figure 3.7b), which make everyday objects smart. Future applications are presented as tags which are applied to retail items as sensor. This way the buying behaviour of customers can be tracked with more privacy by avoiding cameras [Lab18a].

3.3.2. Industrial Internet of Things

The *Industrial Internet of Things* (IIoT) represents the industrial facet of the IIoT technology. Rather than focusing on consumer products such as described in section 3.3.1, the focal point here is the manufacturing and industrial environment. Key aspects of IIoT are technologies enabling machines to operate more independently from human interactions. Paradigmatically technologies are machine learning, artificial intelligence and machine to machine communication. The communication between various machines and also machine components is a key aspect of IIoT.

[1] http://jacobsschool.ucsd.edu

3.3.3. Blockchain Platform for Industrial Internet of Things

[BM16] proposed a utilization of connecting the blockchain with the internet of things called "Blockchain Platform for Industrial Internet of Things" (BPIIoT). In this paper, the authors describe that current IoT technologies in manufacturing are mostly cloud-based, service-oriented manufacturing models. These models have a structure in which users depend on intermediaries for the provision of services. The model presented by [BM16] uses *Smart Contracts* to process transactions and contract modalities (see figure 3.8). A distinction is made here between human to machine and machine to machine (M2M). The cloud based system is supported by a decentralised peer-to-peer network in combination with the blockchain technology.

Thus, it should be possible to offer a marketplace for cloud-based manufacturing services

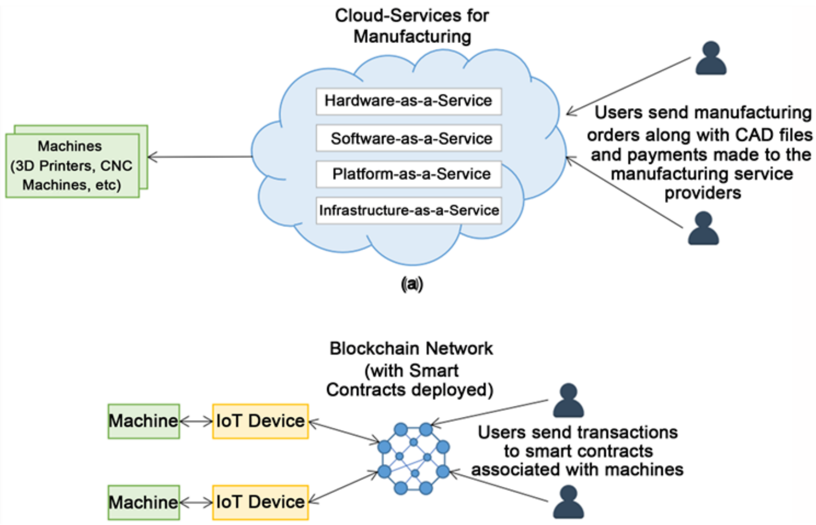


Figure 3.8.: Comparison between cloud based and blockchain based manufacturing [BM16]

by means of the BPIIoT, where users as well as machines have their own blockchain account, all transactions within the network can be carried out without an intermediary and direct order acceptance and processing among the network participants is made possible. It also simplifies the integration of older production facilities into the cloud environment and provides a decentralised, secure, shared general ledger of all transaction data and inventory records. To demonstrate the BPIIoT, *Smart Contracts* were implemented for automated service requirements and spare parts deliveries in the event of a possible machine failure [BM16].

3.4. Decentralised production processes

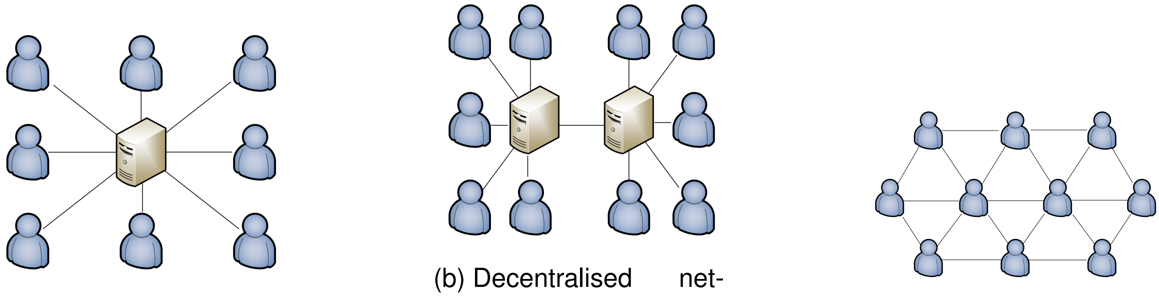
Decentralisation gained a lot of attention with the rise of the term of Industry 4.0. In recent years, the decentralisation of planning and control tasks has always been discussed as an essential component [LA97]. With today's technological possibilities, it is far easier than ever to achieve business goals. Existing technologies, such as RFID or sensor technology, can be used in production in a completely new way, whereby the economic aspect must always be taken into account in order not to make any bad investments in IT systems [Jah17].

3.4.1. Smart factory

A core element of this fourth industrial revolution is the concept of decentralised control using cyber-physical systems (CPS). This is a system in which situational self-control takes place. A constant exchange of information between so-called "intelligent objects" is mandatory. These smart items make their own decisions based on predefined rules during the production process. Theoretically, everything can be an intelligent or smart object. Depending on the complexity of the object it is possible to implement an intelligent aspect. This can reach from a simple RFID tag up until a sophisticated piece of IoT hardware. However, this is not the usual case, as manufacturers of smart products do want to sell various specific products and no individualised solutions that can be applied to any object. From a monetary and marketing point of view this would be fatal.In a CPS, the production resources are distributed spatially. Production areas are combined in a network, but work independently [Bro10].
CPS alone does not provide enough power to run a decentralised production. The communication between the devices and machines has to be assured. The integration of information and communication technology [Fur15] and cyber-physical systems into the production environment creates the cyber-physical production system (CPPS). The combination of communication and the object intelligence forms the so-called intelligent factory (*Smart Factory*). A *Smart Factory* is the ultimate goal of industry 4.0 supported facility [cyb15].

3.4.2. Production process

A production process can be seen as procedure where a certain output is generated in relation to the input. In this elaboration the input is presented as raw material which is



(a) Central network (b) Decentralised network (c) Distributed network

Figure 3.9.: Three different network structures

then crafted or further processed into products at the production site. Having this kind of production cycle, various points are important to be considered. In an economic perspective, the most important thing is to generate the value and also monetary profit as high as possible. Hence, the whole process has an impact on the price it is important to optimise not only the production itself but also all components and departments around.[Sch06]

3.4.3. Central production environment

Centralised systems have been around for centuries. The structure of thus organisations is almost always the same. One single entity has control over more little entities. The failure of just one entity can lead to a total collapse of the production cycle (see figure 3.9a). The fact that more entities are dependant to one higher entity can also be seen as an advantage. In the case of human beings, the higher authority relieves the subordinate entities of responsibility.

3.4.4. AGVs in decentral production fields

Automated guided vehicles are usually controlled centrally. The vehicles only receive information about the transport orders to be executed and execute them, whereby the entire coordination of the orders is taken by a central control system. To not only make the control but also the order management decentralised the central unit as an intermediary has to give way for an decentralised solution [Ull14]. A decentralised method using blockchain technology is being investigated in the course of this work.

4. Conceptual design of a decentralised environment using blockchain technology

In this chapter the knowledge gained about blockchain technology and production processes is put into practice by conceptionally creating a decentralised logistical environment for track based automated guided vehicles. Also different approaches are made in order to fulfil the requirements for an appropriate industrial environment.
First in section 4.1 the background and intent of this project is explained, in order to answer the question what the purpose of this elaboration is. Subsequently the intelligent environment as well as the procedure of the track are described in section 4.2. The role and use of the proposed blockchain is covered in section 4.3 in a single vehicle environment. The case of multi vehicles in the same environment is addressed in section 4.4. The additional use case of applying the blockchain for certifying products is covered in section 4.5.

4.1. Motivation

Traditional supply chains are an interlinking of several entities (see figure 4.1). A failure of just one of the entities will lead to a collapse of the entire supply chain [Pau15]. Preventing the system from collapsing through a single point of failure, it is necessary to decouple the relations among each other. Also the relationship between the entities is based on reciprocal trust. This can be an issue for the existing participants in the supply chain if a new party enters the ecosystem. Every participant within the supply chain is an entity. This can be just one person or an entire company. Recently "Amazon" [1] - the biggest e-commerce company worldwide - exposed delivery services by planting fake packages into the delivery truck deliberately, that got stolen instead of getting returned to the warehouse as requiered by law [Pak18]. Deploying a decentralised supply chain is one possibility to mitigate those kind of risks and failures right from the beginning.

[1] Amazon.com Inc., www.amazon.com

Logistical processes do require a high degree of operative effort [MH18]. The bigger and physically apart the different business partners are, logistical processes get very complex. In cases such as customs, efficient transport, penalties and many more, issues get involved into the process which are carried out by human beings. This brings a lot of bureaucracy and paper work and multi interface communication which makes the processes usually longer and also more expensive. Even more when the supply chain reaches international and overseas dimensions which is very usual in the age of globalisation. It can be said that the more intermediaries get involved into the process the more deviations are likely to happen.
Within this elaboration a concept for a logistical environment is proposed by using the blockchain technology as a tracking tool in synergy with IoT enhanced production segments. The main goal is to investigate the usability of blockchain technology in an industrial and logistical environment to serve as a tool for provenance and tracking of raw materials as well as assembled products. The whole concept will be created under consideration of the *Industry 4.0* and *Smart Factory* approach (see section 3.3 and 3.4.1).

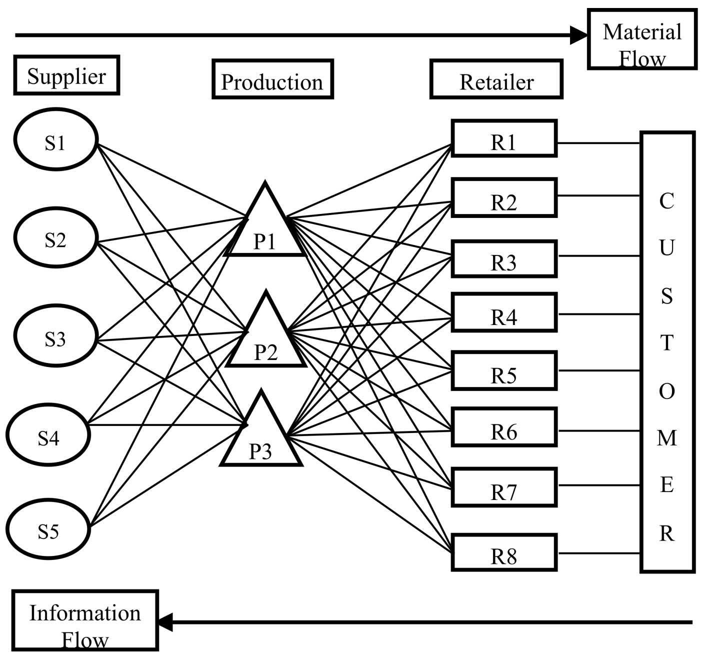


Figure 4.1.: A supply chain system [Pau15]

The focus will be laid on the conceptual implementation of a blockchain as a distributed ledger into the system and will be be scrutinised in terms of its applicability in depth during this paper. AGVs will serve as means of transport for the various goods that are handled. As the most lost of data and goods occurs in the transport between supplier and manufacturer, particular attention will be paid to processes where AGVs are related to. The AGVs will not get orders from higher entities but will be acting as autonomous as possible. This way the operational costs are aimed to be reduced. All of this is approached to be handled in a decentralised business environment. Decentralisation does not only insure an exclusion of a single point of failure but also a high degree of system modularity and flexibility. Furthermore the use of certificates for materials and products is investigated in section 4.5.

4.2. Intelligent environment

There are a few paradigmes to consider while building a facility that is compliant with *Industry 4.0* approach [Rot16]. The goal is to present a concept of a fully autonomous factory environment with the focus on the logistical processes. Hereby, it is intended to develop every entity in the supply and manufacturing chain as autonomous and smart as possible. For the purpose of this study a rather small scale is considered. However, the idea of the concept is able to scale up to any company dimension theoretically.

4.2.1. Vertical and horizontal integration

The first step in developing a successful smart infrastructure and industrial environment, is to lay a solid foundation in which CCPS (see section 3.4) can operate efficiently. However, vertical and horizontal integration do matter in many ecosystems. Figure 4.2 represents vertical and horizontal integration.
The red circle represents the core business. The blue circles represent the vertical integration towards the end consumer, in this case a potential customer of a car. The green circles however, represent the backwards vertical integration that goes back in the supply chain towards suppliers and producers. The orange circle represents the horizontal integration.

Vertical Integration

In the context of *Industry 4.0*, vertical integration means that all company-internal systems are arranged in the same hierarchy. This includes colision free interfaces for flawless and

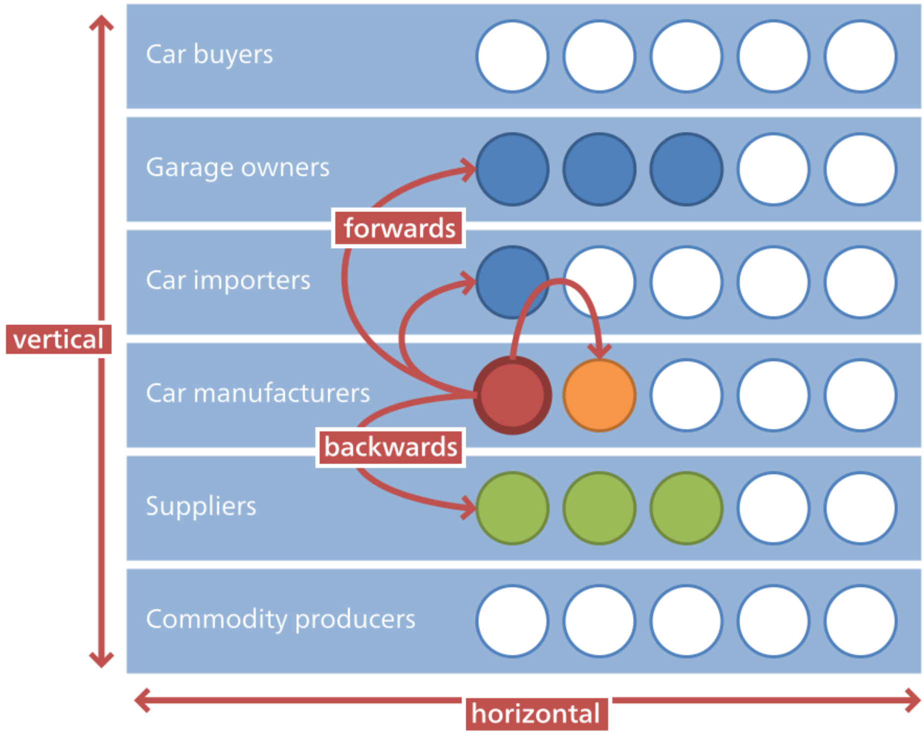


Figure 4.2.: Schematic representation of vertical and horizontal integration in the automotive industry [Sau18]

reliable data exchange between the resulting hierarchy levels. This leads to a uniform and consistent system in which the direction of data must be oriented to the hierarchy order [Rot16]. For a system in an ecosystem that is build up from the ground up, vertical integration is of very high importance. Consistent and possibly open interfaces are advantageous. This way the network can grow faster. For this purpose, the concept will be built upon the "Arduino"[2] platform. Not only is the software open source, but there is a broad variety of peripherals available. Section 4.3 will go deeper into technical and process aspects. However, vertical integration can only succeed, if the communication protocols and interfaces are consistent. To create a successful structure with an integration-friendly environment, proprietary components should be avoided. Using proprietary components or protocols is only an option, if the situation in the market is of monopolistic nature that forces competitors and collaborators to use it [Rot16].

[2] https://www.arduino.cc/

Horizontal Integration

As to observe in figure 4.2, the horizontal integration defines the integration of external entities into the own vertical structure such as suppliers, manufacturers or service providers. The aspect of horizontal integration can be used as a tool for scaling up the company. To be more extensively described, the integration of more suppliers and manufacturers is a factor that needs horizontal integration to be functioning [Rot16]. Having a thoughtfully designed vertical architecture, makes it is easier for other entities to integrate into the system.

4.2.2. Smart Item

The goods that are being handled within this course need to be *smart*. That is a major criteria in order to build a process and supply chain that is working smart and decentralised. It is important to notice that the actual product or raw material that is going to be processed needs to have capabilities to be called smart. Very common technologies are tagging physical objects with data carriers and if necessary, with sensors. Typical and fairly easy data carriers are RFID chips, as they are very easy to install and effective [Rot16].
In terms of manufacturing for instance, semi-finished products with RFID chips can carry data such as following machining processes or overall material condition. The machine then can evaluate the information and adjust the machine parameters accordingly.

4.2.3. Decentralised Autonomous Organisations

Since having smart items in the production cycle, it is comprehensible that the entities that are evaluating and processing the items have to be smart too, in order to ensure a *Smart Factory* environment.
Decentralised autonomous organisations (DAOs) are organisations or businesses, that determine their own decisions, without any external influence. The decisions made, are based on an embedded *Smart Contract*. As long as the conditions do not change in which the DAO is involved in, the DAO can run fully autonomously. The organisation executes its actions based on the embedded code only [Uni17]. The DAO's described in this elaboration do act by the concepts of CPPS (see section 3.4.1).
For the purpose of this study, supplier and manufacturing sites are seen as DAO's. Due to

the fact that suppliers and manufacturers are stationary (the geographical position is firm), the processes of such entities are more or less repetitive after an acclimatisation period. For this, previous human help is mandatory in order to set up the settings of such an DAO.
The majority of the people working in a manufacturing site are doing manual work in the work shops. Nowadays, robots already have a big share especially in the manufacturing and logistics industry. After all, the potential of more automation is not decreasing, on the contrary, the manufacturing and logistics business have the greatest potential for automation after the gastronomy and hosting business (see figure 4.3). Especially in countries with higher population, the potential is very high, as of today still a lot of the work is done manually by humans in those countries [Bra17].

In manufacturing businesses, the majority of the employees do either manual work in the

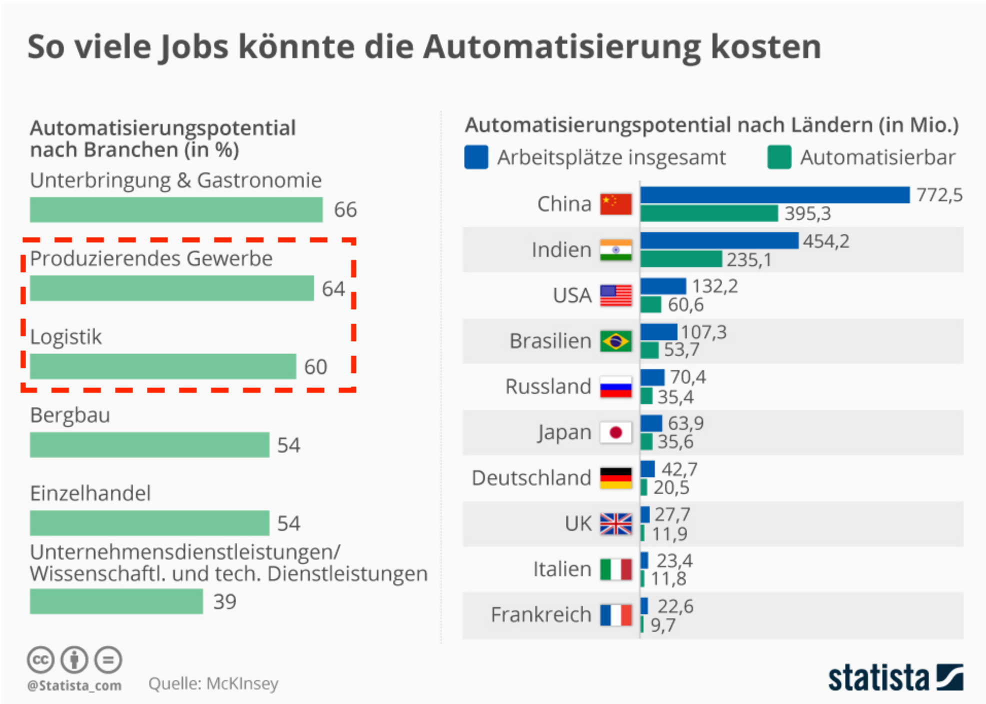


Figure 4.3.: Potential for automation divided in business sectors [Bra17]

production site or operative work on the supplier site. For the manufacturing business we assume that the machines are fully autonomous so that permanent human operation is not needed. On the other end, for suppliers it is assumed, that all processes that affect the materials are done by machines. Also the operative work and communication between other suppliers or clients are automised. As long as the DAO can reproduce the work of the employee it can fully replace it. However, DAO's can help out to enhance the productivity in an industrial process, but they cannot replace all employees in an organisation for administrative and safety purposes [Uni17]. Therefore humans cannot be replaced to this moment for the majority of the strategic and interpersonal work, such as negotiations or inspections at suppliers that operate from different locations.

The first ever DAO is alleged to be the *Bitcoin* network, as it is not controlled by any central

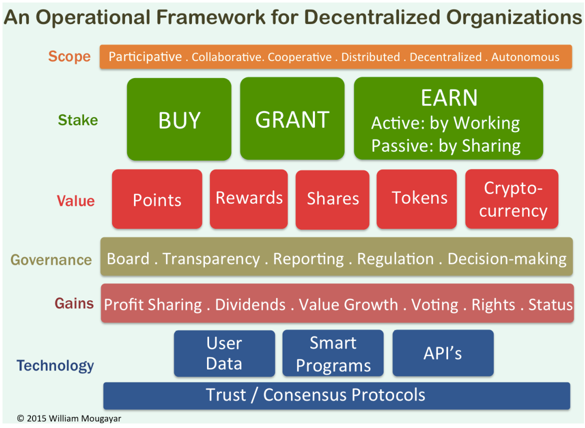


Figure 4.4.: Schematic comparison between conventional businesses and decentralised autonomous organisations [Vos18]

authority, but the distributed consensus only (see section 3.1.4) and is open to the public [Her18b].

DAOs can technically be seen as some sort of a *Smart Contracts* [Vos18] as they execute actions based on a program code (see section 3.1.9). Detached from company structures nowadays, DAOs do not have a central authority that is in charge of the people or other DAOs under them hierarchically (see figure 4.4).

All entities are equal to each other and only operate based on consensus compliant code independently of their role in the company. This fact is very important for a decentralised environment. They basically replace the middle man that executes actions upon actions in conventional companies. This way a lot of time and money can be saved as all decisions are already predefined, whether in conventional organisations a various amount of parties are involved into the decision process.

A very important role of DAOs arises in the negotiation phase in an autonoumously business environment [ZW16]. With the implementation of machine learning DAOs of both sides (sink and source) can learn to negotiate depending on business influencing constraints. For example, a supplier can raise the prices for a certain product, as it knows that no other supplier can provide this kind of material. This way the manufacturing site is forced to buy for a higher price or needs to wait and therefore shut down the production temporarily. The aspect of decentralisation is covered up by using DAOs as organisations without a central authority that controls either the manufacturing site nor the supplier.

Nonetheless DAOs do not come without disadvantages. Once the DAOs are launched based on a *Smart Contract* into the blockchain, none of the deployed code in the *Smart Contract* can be changed. On the one hand this is good, as it is safe against cyber attacks or manipulations by third parties, as long as no entity has control over 51% of the network computing power. On the other hand though, in case of bugs or any needed changes within the framework of the business, it cannot be adjusted accordingly. This gives frauds the possibility to not change the *Smart Contract* itself, but to take advantage of potential loopholes in the code. *The DAO Company*, a start up business founded by two german brothers, got exploited by such a bug letting the whole company collapse. Therefore it is of major importance to launch the *Smart Contract* either with a faultless code or giving a specific amount of DAOs the right to change the *Smart Contract* (see section 4.3.3).

4.2.4. Concept and procedure of the track

For practical purposes a track is build in the form of a model train layout. The mock up of the test trail way has been designed and assembled by Prof. Schelberg [3] and HAW students [4] as part of a university project collaboration. Figure 4.5 represents the track that is being built. Furthermore, the process for a single vehicle situation is illustrated by arrows. So far the rails have been setup. The following scenario describes the case, how the model should work in the future. The operating energy for all components will be provided by an external power supply.

The model consists out of three suppliers and three manufacturers. As this is just a model, this does not represent real world conditions. For demonstration purposes, the dimension of the track is held efficient and portable. However, the concept shown is not only applicable for intra-logistical or short distance tasks. The model also can be applied to nationwide even

[3] https://www.haw-hamburg.de/beschaeftigte/detailansicht/name/hans-joachim-schelberg.html

[4] Alexander Bokow, Jörn Hermann, Benedikt Zimmermann

international logistical trades in reality. Figure 4.5 shows the concept of the track.

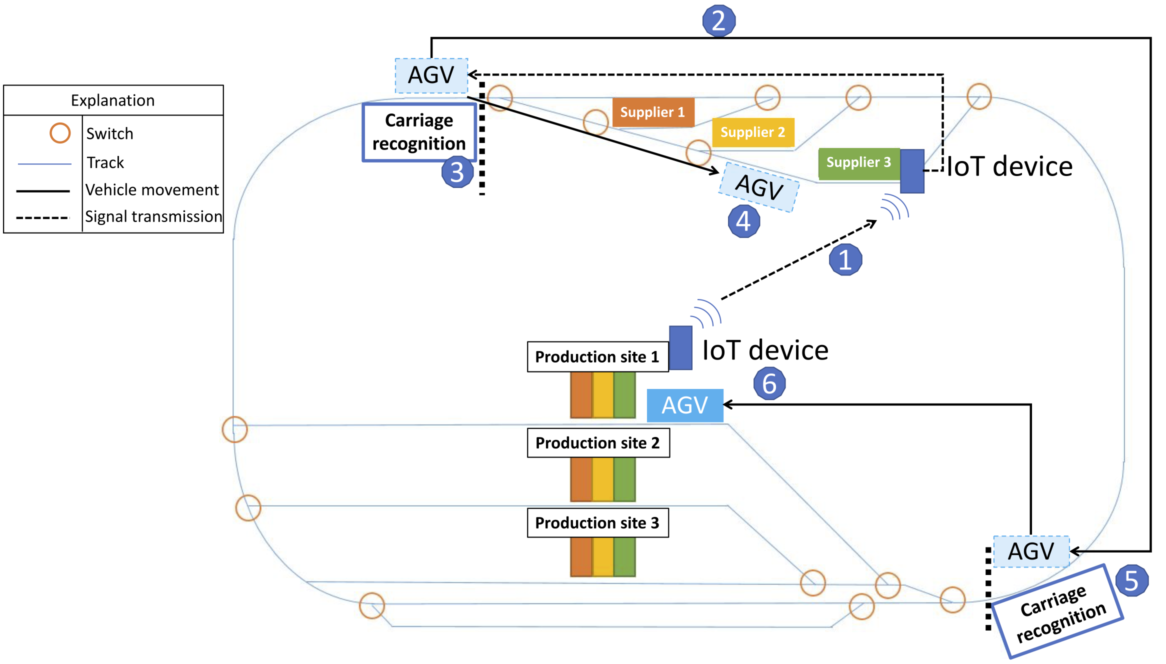


Figure 4.5.: Technical procedure of order triggering and delivery

The general procedure of the concept is, that the manufacturers produce goods for which they need material. The material is provided by the supplier. The components in the manufacturing facilities as well as the supplier facilities are equipped with IoT-supported hardware (CPPS), motivated by the *Smart Factory* approach in section 3.4.1. In Figure 4.5 only one IoT device is illustrated for each end. This is for demonstration purposes only. In reality, every entity needs to be equipped with at least one piece of IoT hardware in order to be considered smart. Usual cases would be, that single machines are implemented with various IoT hardware for different purposes. Depending on the IoT device various indicators can be monitored. Different hardware monitors different indicators. It is assumed that the IoT hardware monitors the machines and overall processes in the production facility. On the supplier side the IoT hardware monitors the stock and warehouse including the inventory.
For creating a decentralised environment the approach is to design the entities autonomously without humans being directly involved into the process, after the adoption phase. The overall procurement process is displayed in figure 4.6. The exact process will be elucidated in

the following sections. The explanations will refer to figure 4.5. The goods are tracked by the *HAW Coin* [5]. The detailed structure of the transactions and coin properties will be discussed in section 4.3.1.

Step 1: Material shortage

The manufacturing site is producing products out of raw materials. Each material is part of a final product. However, the end product contains different proportions of the raw material. The different colours in figure 4.5 represent the different materials, respectively the corresponding supplier (orange, yellow and green).
If there is a shortage or a deviation in the manufacturing plan, the affected IoT device in the production facility is triggered. The IoT device is triggered not only in case of existing material shortage, but also in case of a sudden need of materials for future products or for orders that came in short gritted. The fulfilment of short gritted or manufacturing of low batch sizes are an advantage of decentralised manufacturing systems.
The IoT device sends out a message to the supplier that provides the requested material. The message is received by another IoT device on the supplier end. The message that is sent, includes an order in the scope of eliminating the deviation in the production and prevent downtime in the manufacturing process. Hereby, the manufacturer communicates with the supplier via a *Smart Contract*. The embedded *Smart Contract* in the supplier structure contains all sort of information for production and material. Possible deviations can be material shortage, worn material, shortage of operating resources and much more. Within this framework the scarcity of materials and their procurement is being focused on.

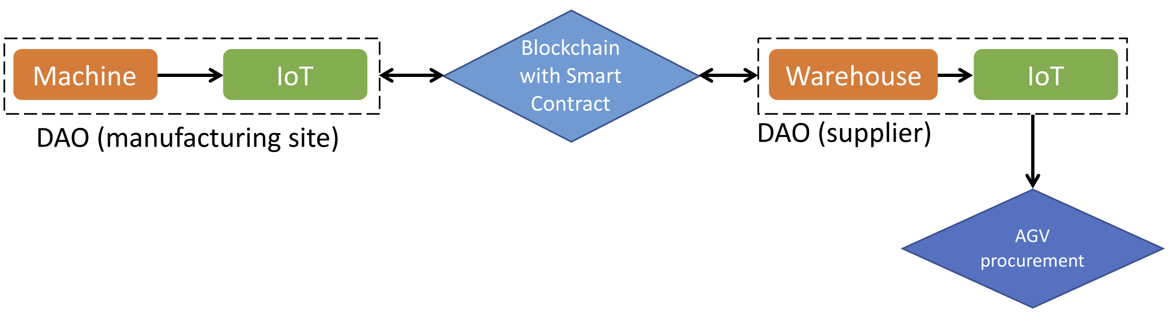


Figure 4.6.: Schematic view of the process

[5] fictional name of the custom tracking coin created for this project

A production site can also order material in advance, if corresponding data in the *Smart Contract* do indicate to that. For example, if one of the suppliers does not operate in the following days. Foresight in production processes reduces machine downtimes and storage costs. The communication between supplier and production takes place directly in order to maintain the degree of decentralisation and lack of intermediary. To this end, each entity must maintain an active internet connection at all times in order to be part of the network. An exclusion from the internet also means an exclusion from the system. Furthermore, each DAO stores a copy of the blockchain and act as a node. That way DAOs verify whether a transaction is valid or not. This is part of the consensus the system is build upon (see section 3.1.4). Once the supplier has received the order from the manufacturing site, the order is being processed.

Step 2: Vehicle request

In order to deliver the material to the production site, it is necessary to deploy a vehicle, which first picks up the material from the supplier and then transports it to the production site. Switches are installed prior to the gateway to those areas. The AGV receives the information wirelessly from the suppleir. The information consists of path details and further material details, such as amount and weight. This received information is written on the RFID chip, that the vehicle is carrying. The principle of how to receive wireless information and write it to RFID chips on the base of Arduino, is practically explained in [Wen15].
For simplicity reasons the initial conceptual phase considers an environment with one AGV only. Hence, no traffic exists on the tracks with only one vehicle operating, the AGV directly finds its way to the sinks and sources without downtimes or detours.

Step 3: Vehicle recognition and switch control

Having arrived in front of the switches for material pick up, the information saved on the RFID chip is evaluated by the carriage recognition sensor. As there are three switches build in series, there is a variety of eight possibilities for the combination of three switches. Every switch can adopt two positions.
Figure 4.8 shows in which direction the vehicle drives, if a switch is positioned in a certain way. From figure 4.8 it can also be observed in which direction the vehicle drives. A switch

can either have the position "0" or "1" as it is motorised by a two way servo. The servo communicates with the carriage recognition directly.

circuit diagram				
	Switch 1	Switch 2	Switch 3	Destination
Path 1	1	1	1	*Neutral*
Path 2	1	1	0	
Path 3	1	0	1	
Path 4	1	0	0	
Path 5	0	0	0	C
Path 6	0	0	1	C
Path 7	0	1	1	A
Path 8	0	1	0	B

Figure 4.7.: Circuit diagram

Figure 4.7 shows the combinations in the case of three switches aligned in series. The paths from 1 to 4 all do not lead into the area of the suppliers, but to the main track as the first switch is on "1". On the other hand, the highlighted paths 5 to 8 do lead to the section where the three suppliers are located. However, it is to be noticed that not every path does need three switches to operate in order for the vehicle to reach the correct destination. Path 5 and path 6, which lead to *Supplier C* only need two switches to operate for assuring the vehicle to get to its destination. Under model conditions and with one vehicle only it does not make a big difference if all three switches are operating, although it would be enough to only operate two switches for the same result. In a scaled up environment with a high frequent rail traffic though, the wear of the switches (especially mechanical components) as well as energy consumption are definitely aspects to take into account for a real environment. Path 1 is here declared as *neutral* with all three switches having position "1".
Due to the fact that the track where the manufacturer facilities are located is build up equally, this approach is valid analogously for the junction on the manufacturer side. The control logic remains also the same. Figure 4.8 shows the visual representation of the circuit diagram. The numbers "0" and "1" on each side of the rail indicate in which direction the switch tunes.
At each switch there is a servo which is responsible for the position adjustment. The servos receive the information of the switch position from the carriage recognition with which they are connected (see figure 4.5). The carriage recognition reads and validates the information from the vehicle. The information validated by the carriage recognition, consists out of one of the paths the vehicle has to take for reaching its destination (see figure 4.7). According to

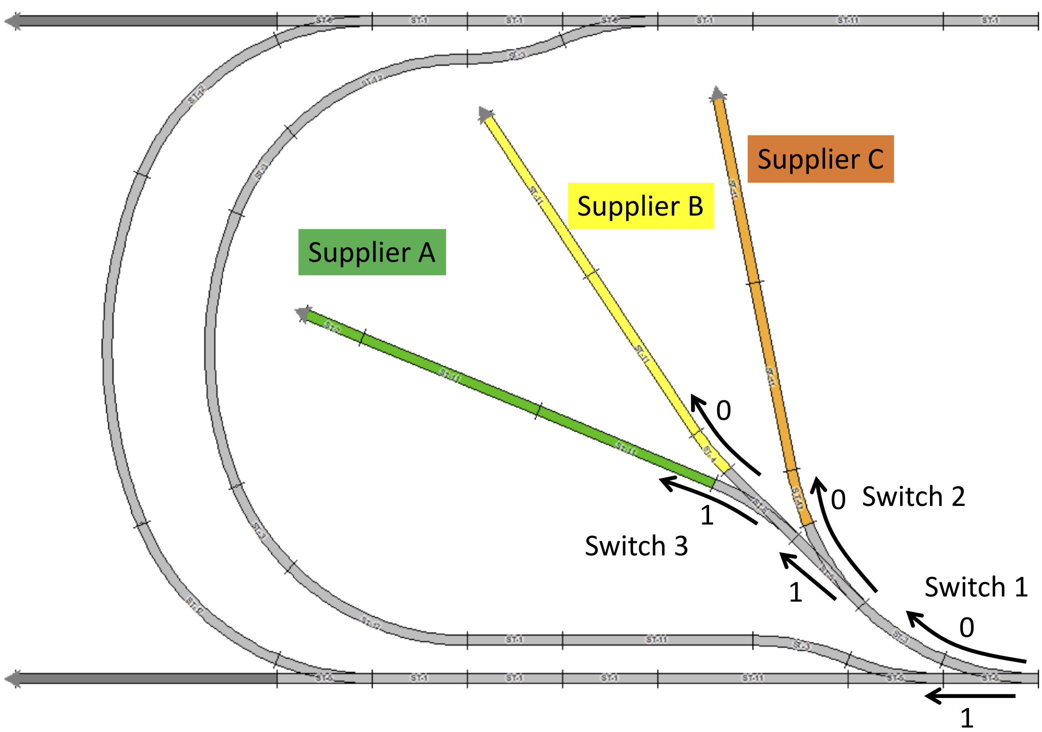


Figure 4.8.: Possible paths for several switch positions on one half of the track

the path stored on the RFID chip, the servos do adjust to the according positions in order to fulfil the corresponding path for the vehicle.

Step 4: Vehicle deployment

Once the vehicle has reached the supplier, the material is charged on to the loading platform of the vehicle. At the same time, the information about the of material and the destination information are stored on the RFID chip of the vehicle. The transfer of the tokens from the supplier to vehicle us detected as a transaction on the blockchain. As the supplier propagates this information to every entity in the network, each party in the system has knowledge about the transaction. Also information on how much of which material has been loaded on to the vehicle and where the material is to be delivered from the vehicle. The stationary entities act as nodes in the blockchain. More detailed explanation on the technical side will follow in the sections below.
The access rails to the suppliers and production sites are dead ends. For that reason, the vehicle needs to push back to the main track. The time from the vehicle entering the section

where it has access to the suppliers directly, the switches do not change. This assures that the vehicle can drive backwards out of the junction.

Step 5: Recognition for destination arrival

After having all the material loaded the AGV finds its way to the supplier that requested the material in the first place. Similar to step 5 the information on the RFID chip is being read by the carriage recognition system. As the information of the destination is stored on the RFID chip the switches are shifting accordingly, so that the vehicle can proceed to the right supplier. The circuit diagram for the switches is identical to the representation of figure 4.7. The only difference here is, that now the vehicle is approaching manufacturing sites instead of suppliers.

Step 6: Arrival and unload

The requested material is unloaded at the corresponding supplier. This procedure triggers a second transaction on the blockchain. Again the transaction is propagated into the blockchain. In dependency on the initial transaction when the material got on the vehicle, this transaction is validated by the other entities of the supply network. It has to match exactly, otherwise it is definite that something went wrong during the transportation process. If the order allows deviations, the deviation cannot exceed the pre-defined tolerances. This applies more to material that is valued by weight only and not in pieces.

4.3. Implementation of the blockchain

After having created a detailed overview over the environment and the processes of the system in section 4.2, the blockchain shall be developed and implemented into the process described above. Terms like *Token* or *Smart Contract* already got mentioned in the course of this paper. In the following the knowledge will be applied to the concept of a logistical environment.

The integration and implementation of a blockchain into a new developed environment needs to meet certain conditions. As the logistical system shall be a decentralised model, one aspect to consider is that the ecosystem that is being developed is as versatile as possible. This gives the model the possibility to grow not only physically, but also for making the growth of the network as big as possible. Deploying a blockchain ecosystem that is not used by anybody would make no sense. As blockchain already has a lot of attention, new blockchains would suffer from early cyber attacks and high vulnerability. Also a high number of participants are making the network more safe and more resilient against attacks. Therefore in the following aspects are accounted for assuring just that. For using the blockchain as a distributed ledger, every stationary entity needs to have the exact same copy of the blockchain. These entities are also called *nodes*. This counts for the suppliers as well as for the manufacturing sites. The nodes verify the state and trueness of the transactions. Hence, during the explanation of the blockchain implementation, it is considered of having one supplier and one manufacturing site in the beginning for simplicity reasons, also. This way a solid base is build regarding the vertical architecture which can later be followed up by adding the entities and nodes as part of an horizontal integration (see section 4.2.1). For building the vertical architecture for the logistical environment a couple of things need to be noticed. One aspect is to define the structure of the stationary entities. Hence, not the whole supply chain is taken into consideration but only the logistical aspect, end consumers will not be involved in this observation. However, end consumers cannot be neglected as they are a significant part of the supply chain.

4.3.1. Blockchain type

As discussed in section 4.2.1, a well build vertical architectural environment defines the integrity of a product. This ensures that the idea which is developed does not only serve oneself, but has potential for scaling. A highly regarded system is not only used by a lot of people and companies but in case of blockchain technology more participants are making the network stronger. This is key to a successful and sustainable business. The blockchain technology that is going to be used throughout the project is one key aspect to consider. A selection of a technology on the off chance could have irreversible consequences that would make the whole ecosystem obsolete. In the following the selection of a blockchain with its

extras such as side chains and second layers is made.
The vertical structure of creating a blockchain starts by choosing the type of blockchain that is going to be used. The first step before implementing a blockchain besides choosing the blockchain type is to define the architecture and the functional mechanism. The blockchain variations and characteristics are comprehensively characterised in section 3.1.2. The purpose of the blockchain is decisive for the type of variation, therefore a short designation is made on what and by whom the blockchain will be used.
The main purpose in this elaboration is to track material during the production and along their product life cycle. The payment aspect through the blockchain will be discussed secondarily. The most common existing blockchain networks are the *Bitcoin Core Network* [6] and the *Ethereum Network* [7]. Currently both networks base on the *Proof of work* (see section 3.1.7 concensus algorithm.[Flo18] However, the *Ethereum Network* wants to step away from a *Proof of work* mechanism into a *Hybrid Casper* architecture FFG(Friendly Finality Gadget) which is proposed in in the *Ethereum Improvment Proposal* (EIP) 1011 [Flo18]. Since the blockchain shall be used by corporations with potentially confidential data, security of the data is a big factor. As the data shall be as safe as possible it is proposed to build the upcoming system upon the current [8] most secure blockchain network. Hence, the decision is made in favour of the *Bitcoin Core Network*. The hash rate per day of the *Bitcoin Core Network* is currently the highest in the blockchain space[9]. On the other hand, the *Bitcoin* network is a public permissionless blockchain. This makes the data stored in the blockchain transparent to the public. (compare section 3.1.2). This trade off shall be accepted as it is not personally apparent, which entity posses how much of goods. The data gets public once a transaction is made.
As *Bitcoin* is a crypto currency it is not eligible for using it as a tool for tracking assets such as material in a manufacturing and logistical environment. Yet there are existing technologies which can be used on top of the *Bitcoin Network*. Most of these technologies work as a *sidechain* or as *second layers*.
Comparable to the first years of the internet there were not many protocols available. The internet itself was the base where various protocols were build upon for different use cases

[6] https://www.bitcoin.org
[7] https://www.ethereum.org/
[8] current date: 23th October 2018
[9] https://bitinfocharts.com/de/comparison/bitcoin-hashrate.html

[Don18]. This concept exist in the crypto space as well. Just like TCP/IP which is a protocol build upon the base of the internet, different layers and side chains do exist for crypto currencies. Unlike proprietary protocols like AppleTalk[10] or NetBEUI [11] TCP/IP is an open protocol that is build modular. As the *Bitcoin* network is equally open and independent as TCP/IP in the internet space it will be used as a base within this framework.
Side chains and second layer protocols are technologies that build upon the main protocol to enhance it by contributing functionalities that the main network cannot provide. Side chains enhance the existing blockchain. One side chain application for example is the use of *Smart Contracts* upon *Bitcoin* as the *Bitcoin* base protocol is unable to run *Smart Contracts*. A couple of side chain and second layer concepts will be presented in the following.

Coloured coins

Coloured Coins[12] are not actually colored. It is a term to highlight the customisable feature of them. With *Coloured Coins* it is possible to create unique tokens with meta data embedded into it. The coins can still be used for payments. To create *Coloured Coins*, a certain quantity of *Bitcoin* is reserved for a specific use. Each use case is given a different colour. The reserved coins are then inserted into a digitally signed "color definition" along with meta data, such as weight, age or physical dimensions. Once a colour is allocated it cannot be changed. Any colour-conscious client that receives this coloured definition will assign the desired meaning to each coin that comes from the specific issue. Within this framework this means that every material would get explicit colour attributed.[Ros18] In order to be able to trace coins back to their origin, a set of rules is defined which allows to preserve, in an unambiguous way, the color of coins moving in a transaction. The rules guarantee that in every transaction, the total output value of coins of any color is no greater than the total input value of that color. Thus, every party can only send coloured coins given to it, it cannot create new ones [Ros18]. This is especially beneficial for fraud prevention. It has to be outlined that not a whole *Bitcoin* is needed to transform into a coloured coin. Every *Bitcoin* has 100.000.000 subunits, that are called *Satoshi*[13]. Each *Satoshi* can be coloured

[10] Apple's proprietary internet protocol
[11] Microsoft's proprietary internet protocol
[12] https://en.bitcoin.it/wiki/Colored-Coins
[13] 1 Bitcoin (BTC)=100,000,000 Satoshi

separately with another colour. This gives this concept a very big variety in usage of the subunits.
Disadvantageous to this concept is that, if a coin is coloured it cannot be changed afterwards. This makes sense as if it would be possible to change the possession of the coloured coin. Another disadvantage from a industrial point of view is that if the corresponding material gets discontinued, the coin basically becomes worthless.

Lightning network

Another emerging layer technology is called *Lightning network*. In contrast to *Coloured Coins* this layer dedicates towards the scalability issues of the *Bitcoin* network. The approach of the *Lightning network* is to shift transfers from the main network to a second layer that works *off-chain*. Two parties agree on opening a channel where both parties store coins on. The channel can be closed at any time by anyone. The money stored by both parties serves as balance. Every time one of the two parties makes a transaction the amount of money is deducted from that balance. At the time of the closure of the channel the cleared balance between the parties is transferred as one transaction into the main blockchain. This reduces the network utilisation and boosts the transaction frequency of *Bitcoin*. The network of financial service provider Visa Inc.[14] *VisaNet* can handle approximately 24.000 transactions per second [Rau18]. The *Bitcoin* Network on the other hand can handle around 7 transactions per second.[CDE+16] With the implementation of lighning network the number could exceed 24.000 transactions per second by far [**?**]

4.3.2. Coin management

In section 4.3.1 it was presented in detail that the *Bitcoin Network* is the most qualified blockchain to use due to the safety and resilience aspects. However, it is not possible yet to launch a custom token based on the *Bitcoin Network*. Currently, *Bitcoin* applications are limited to financial purposes only. That is why for practical reasons the "*HAW Coin*" is launched upon the *Ethereum Network*. *Ethereum* allows to build own tokens, based on its blockchain network. The programming language that is used for programming the blockchain as well as *Smart Contracts* is called *Solidity*[15]. Listing 4.1 shows the constructor of the blockchain.

[14] https://www.visa.com
[15] https://solidity.readthedocs.io/en/v0.4.25/

For actually launching the *HAW Coin* into the blockchain, it is necessary to own a couple of *Ethereum*. As this is not the case, just a model of the code is presented in order to get a feeling in how to actually deploy your own crypto currency.

Listing 4.1: **Solidity Constructor Code for launching an Ethereum Based Blockchain**

```
contract HAWCoin is EIP20Interface {
    uint256 constant private MAX_UINT256 = 2**256 - 1;
    mapping (address => uint256) public balances;
    mapping (address => mapping (address => uint256))
    public allowed;
    string public name;          //HAW Coin
    uint8 public decimals;       //seven
    string public symbol;      //HAW
}
```

In the last three lines the settings for the crypto currency are implemented. *Public name* is the name of the coin itself. In this case it is named *HAW Coin*. *Public decimals* defines the amount of decimal places the currency implies. *Public symbol* is the abbreviation, how the currency is declared. The payment as well as the trust of the *Ethereum Network* is borrowed for this purpose. Payments are done by using *Ethereum* as a crypto currency (see section 4.3.1).

The imaginably launched *HAW Coin* is used for material and product tracking. The advantage of creating a new token is, that it can be tailored as it should be. In this case it is determined, that in total 200.000 coins will be released. That does not mean that the 200.000 coins will be available from day one. They need to be *mined*. Every time the supplier is giving out material, new coins are released into the system. This coins then circulate in the ecosystem. Every coin is subdivided into 1.000.000 (one million) units. That means that one *Mega HAW Coin* equals to 1.000.000 *HAW Coins*. This gives the system flexibility and a prospect for a longer use. As the coin is going to be used for tracking purposes, it has to be matched with the material that needs to be tracked. This of course has to be done individually. One *HAW Coin* can equal to one kilogram of iron, or 1 tonnes of wood. It has to be assured, that the ratio between the material and the amount of coins that is being backed with, is not out of scale. In detail this means, that it shall not be possible to acquire as many coins as possible, by producing or purchasing the cheapest commodity. A small amount of a valuable commodity has to be backed by as many coins as a big amount of a cheaper material. Manufacturing sites and suppliers will act as nodes that evaluate the transactions between

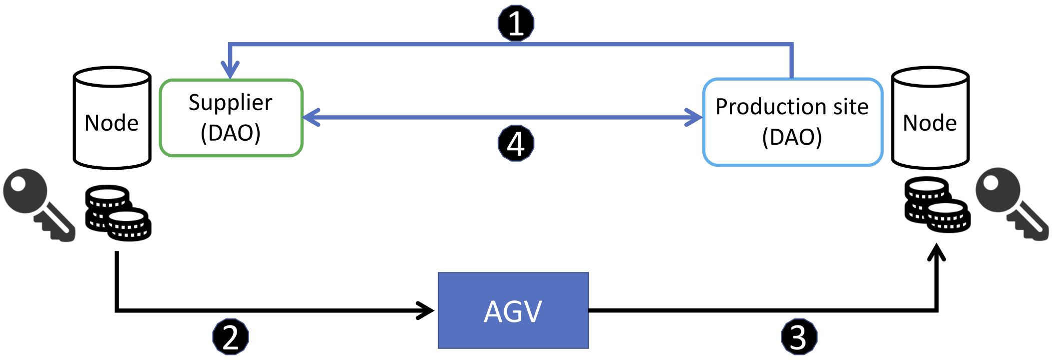


Figure 4.9.: Schematic view of the transaction sequence

each other based on the predefined consensus. Also the evaluation follows the rule of the *Proof of work* concept (see section 3.1.7). This way each transaction needs to be confirmed by the other entities to count as valid. Hereby, the message is encrypted by the supplier, so that the message can only be decrypted by the addressee, in this case, the manufacturing site (see figure 4.9).

Sinks are generating orders which are then send to the sources. The AGV then takes the order and picks up the materials from the matching source. That moment the AGV picks up the material, a transaction is triggered in which the material changes the possession from the source into intermediary possession of the AGV. For preventing possible attacks on the AGV, a penalty is needed, if an AGV gets lost or stolen. As the AGV itself does not have a copy of the blockchain it serves as a transportation unit only. As the vehicle would have had to store the whole blockchain. Furthermore the blockchain size increases with every transaction. In environments with high transaction frequencies, this would lead to a massive data storage, that is always on the move. More data storage means also more weight and less space on the vehicle. The coins are transferred via the RFID chip that the AGV carries. Every entity that conducts transactions needs to have a public and private key (see section 3.1). The coins are transferred to the public key of the AGV. To automate the processes and also to handle transaction of tracking and monetary coins, a *Smart Contract* will be implemented into the process which contains all information and rates referring to the payment.

From the point where the transaction is conducted, the information is propagated to all other nodes in the system. This way everyone in the network has knowledge on possession of the coin and can track the goods that are transported. The time between the transportation from supplier to manufacturer, it has to be assured that the material respectively the coins do not change ownership. That is why it is suggested to put check points along the track. These check points the condition of the AGV load and guarantee that nothing goes wrong along the way. As the check points do not change their positions it is proposed to implement geographical information into these check points. When an AGV passes by, the location is open to see for everyone in the network along with the load condition.
Arriving at the destination the goods are unloaded and the tokens are transferred as well. If everything went correctly, the amount of coins matches with the initial value. Furthermore, the goods need to be complete as well. Since the AGV has put effort into this procedure by expending not only time and providing the service of transport, but also using energy for powering its electrical engine it is entitled a reward. The reward is paid in form of crypto currency in this example *Ethereum*.

4.3.3. Embedding of Smart Contracts

For upgrading the process quality of the concept, *Smart Contracts* can be very helpful. Reviewing the knowledge of section 3.1.9 we know that *Smart Contracts* need to be carefully coded in order to have them operate the right actions. The core structure of *Smart Contracts* is very basic as it uses "when-then" commands. Once a *Smart Contract* is deployed into the blockchain, it is not possible to change it. The hard coded content is executed by the organisation to the smallest detail which makes actions irreversible. Therefore the content needs to be discussed comprehensively by all involved parties. For this human help is imperative. In the following a *Smart Contract* is presented for the environmental conditions that got defined during this work. In section 4.4 an extension for a multi vehicle environment is presented.

The *Smart Contract* serves as a connecting link between the entities. Every interaction between entities that involves transactions, triggers a *Smart Contract* that is deployed in the blockchain network. However this is not obligatory, but encourages the aspect of an automatised industrial environment running on the blockchain. The IoT equipped machines and components are constantly communicating with the blockchain and therefore also with the *Smart Contract*. In the case of material shortage or deviation, that is detected by IoT hardware, information is transfered to the *Smart Contract*, which initiates an action based on the code embedded. In this case the *Smart Contract* contacts the supplier to deliver material to the manufacturer. The supplier in turn then contacts the AGV for material procurement via the *Smart Contract*.

Smart Contracts are an essential component of *Ethereum*. Hence, there is no need of additional second layer protocols or side chains. In case of the use of Bitcoin, which is theoretically capable of *Smart Contracts*, a side chain implementation would be inevitable. Tadge Dryja from the Massachusetts Institute of Technology found a way to run simple *Smart Contracts* on the *Bitcoin* blockchain [Her18a].

As already pointed out, *Smart Contracts* cannot be changed as they are part of the blockchain. However, if a group starts to over think the processes of a *Smart Contract*, a new one has to be deployed. Such as in cases, when new products or entities enter the ecosystem.

4.3.4. Proposed IoT enhanced blockchain environment

The model presented in the following is a concept which is tailored to the previously explained logistical application. It has to be considered that this is not an ideal solution, but a first approach to integrate the blockchain technology into a technical and logistical environment with the inclusion of automated guided vehicles. The architecture that is being presented, represents the basic framework of the whole environment. It can be seen as the representation of the vertical architecture for the environment that has been discussed comprehensively. The synergy between IoT devices and blockchain technology is presented. Also DAO's are taken into consideration as entities operating in the environment. Figure 4.10 represents the architecture of the decentralised IoT environment that has been developed in the course of this paper. The architecture is separated into four general frames that are going to be clarified in the following.

Transaction mode:

The transaction mode is such mode, were all the tracking and also payment is being realised. The *Ethereum* blockchain is used as it the most applicable one for this kind of environment to this time. The straightforward implementation of *Smart Contracts* makes it very easy. For tracking purposes the newly launched *HAW Coin* is deployed, hence *Ethereum* itself is not able to delivery those information. The *HAW Coin* will track material characteristics also. The bidirectional conncection between the *Ethereum* coin and the *HAW Coin* with the *Smart Contract* shows, that every movement of each coin is registred by the *Smart Contract*. Furthermore the *Smart Contract* is eligible to move the coins.

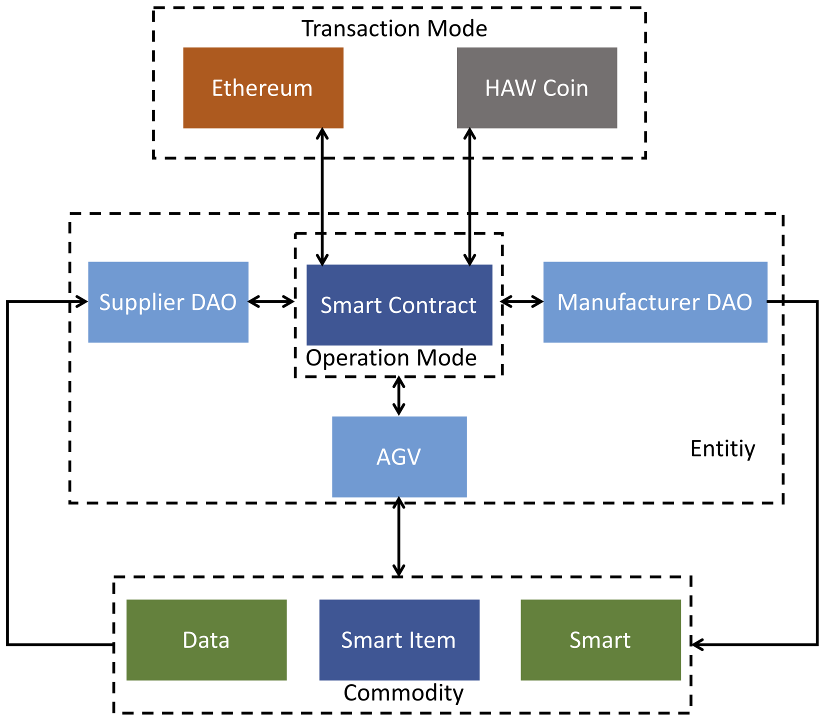


Figure 4.10.: Arcitecture of IoT implementation in supply chain management [BM16]

Operation Mode

The operation mode here consists out of the *Smart Contract*. As described previously, the *Smart Contract* is the interlinking part between the entities and responsible for conducting the transactions between the actors in the network. Referencing the approach of [BM16] the *Smart Contract* controls the activity of all peers either directly or indirectly (see figure 4.10). The commodities here are controlled indirectly, as the commodities directly linked to the facilities (supplier and manufacturer) as well as to the AGVs. The *Smart Contract* contains all data and information that is mandatory for executing the actions between the parties.

Entities

So far entities include only one unit of each peer. One AGV, one supplier and one manufacturer. They all interact through *Smart Contracts*. The communication through the *Smart Contract* assures that the entities cannot do business without the blockchain keeping track of it. With the addition of more suppliers, manufacturers and AGVs, the number of *Smart Contracts* also rises.

Commodities

The commodities in this case is the material. The material has to have certain features to be *smart*. Within this elaboration, the smart component is provided by an RFID tag, where specifications are stored on. With those specifications, the machines can evaluate, how to handle the goods. Details on how smart the commodities are, are explained in section 4.2.2.

4.4. Multi vehicle environment

The vertical architecture was presented in the past sections. However, having only one vehicle in an industrial environment is highly unlikely. Industrial and logistical applications are involving more than one vehicle. The more vehicles are operating in a limited space, the more the level of complexity rises [LAUP11]. Also, the safety requirements do rise as more vehicles and potentially humans are operating in the industrial environment. The work space the AGVs are operating in, is fully known by the vehicle so no learning or referencing processes are needed. Hence, we are dealing with a track based system the routes are also fixed and not changeable.

4.4.1. Material procurement

Considering only one vehicle is fairly easy as there is no intra-communication and coordination. The supplier only has the option to engage with one vehicle only. With multiple vehicles, the whole behaviour in track control changes, whereas the blockchain subjects do not, except for more data. More vehicles do mean more rules, hence, the time efficiency is linearly dependant on the number of AGVs in the system [LAUP11]. Therefore, it is inevitable to launch rules that define how the vehicles behave within the rail system.
A a first rule, it is to work out, which vehicle gets the order from the supplier, that was originally triggered by the manufacturing site. Delivering orders, is the job of such a vehicle. Without delivering any orders, the AGVs will not get anything in return for maintaining their service. This is the case even more in a competitive environment. As every vehicle receives a reward for its expenses and service for transport every vehicle is intrigued to get as many jobs as possible. Figure 4.11 shows a flow chart, that shows the process of assigning vehicles for an order that has been propagated into the system. The figure addresses the communication between vehicle and supplier. The flow chart is a virtual representation, how a *Smart Contract* can work in an intelligent environment. The AGV request is triggered by the supplier. The supplier on the other hand, gets a notification of material request from the production site. This is indicated and communicated by the IoT devices. To find a vehicle that is available and also capable of the job, it is proposed to undergo a procedure for vehicle procurement.

AGV request

The initiating instance is the actual request of a vehicle. This action is triggered by a material request.

Is an AGV available?

The supplier needs to retrieve a vehicle in order to fulfil the request of the manufacturer. Before actually retrieving a vehicle, it has to be assured, that a vehicle has an available slot. In case of no availability of vehicles for a certain time frame, another request is then to be conducted.

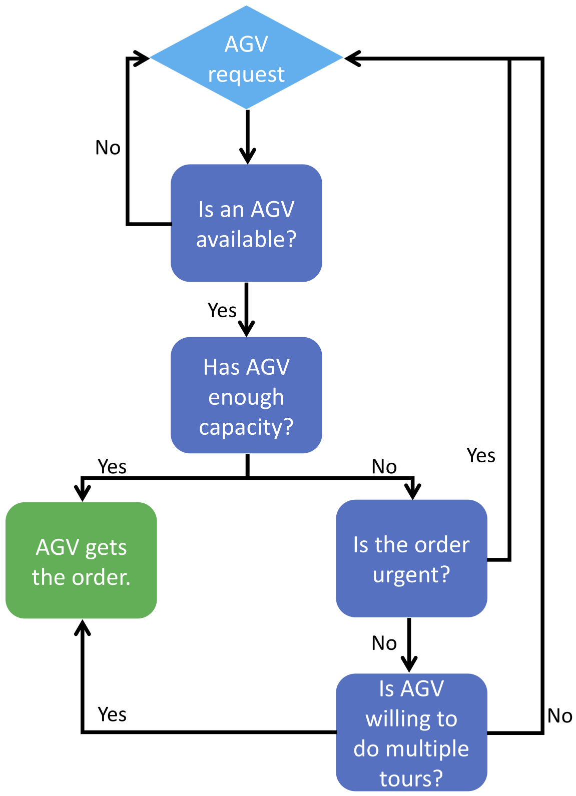


Figure 4.11.: Procurement flow chart

Has the AGV enough capacity?

Once an available vehicle has been registered, further criteria must be applied to ensure that the vehicle is capable of fulfilling the job. As not every vehicle is of identical build, the capacity varies as well. If a vehicle has enough capacity the vehicle will get the order. In the event of a negative response, the level of urgency needs to be determined.

Urgency

Is a vehicle available, but has not enough capacity the question of urgency is needed to be clarified. This information is given to the supplier by the manufacturer at the time of the material request. The urgency of the material is dependant of the degree of necessity to the production site. If the request is urgent it is necessary to find the vehicle with best availability. In the worst case, s new vehicle needs to be requested until an appropriate one is found.

Handshake between vehicle and supplier

When the request of material is not that urgent to the supplier, still there is a question to clarify. The vehicle has the possibility to conclude the request in more than one tour. This gives vehicles with smaller capacities the possibility to fulfil orders and compete with other vehicles in the system.

4.4.2. Award of contract

As in this part a multi vehicle environment is considered, more than one vehicle can match the conditions queried above. The process of which equally qualified vehicle gets awarded with the contract by the supplier, has to be neutral so that manipulations are of low to none significance. A typical method for award of contract is to assign the nearest vehicle possible to receive the order. There are a various amount of methods in existing that are dealing with this topic such as the k-nearest neighbors algorithm [SRI18] or the Dijkstra algorithm [Vel18]. This methods however, aim to solve routing problems by optimising the distance that the vehicle should take. As described in section 4.5 by figure 4.11, vehicles can accept inquiries even if they do not have enough capacity or are further away. The decision is made by the vehicle itself and is independent of the location.

In this concept, all available vehicles get a notification from the supplier if an order is triggered. Similar to the *Proof of work* concept, the qualified vehicles are competing against each other in order to get the job. Throughout the process the procurement decision tree is being run through. To keep things simple in the first stage, only one request at a time can be processed by the suppliers as well as from the vehicles. Amongst all the AGVs which are eligable to get the contract according to figure 4.11 are competing against each other by solving a mathematical puzzle.

For intra-logistical applications however, this concept is not considerable. Within one company there is no competition and also no reward in the means of getting paid for certain tasks. For intra-logistical purposes it is recommended to process the actions as fast and also as efficient as possible.

4.4.3. Deadlock resolution

Deadlocks are situations in which at least two vehicles are hampering themselves. In such a situation, movements of either vehicles are not possible any more. Regarding to the example of the test rail way shown in figure 4.12 we are dealing with three dead ends at each side of the test track. In normal cases, AGVs cannot move backwards. Even less if the vehicles are rail based. A push operation is highly risky for the vehicle, especially in curves. The risk rises when the vehicle is loaded. Even assuming that the AGVs could move backwards, a situation as in figure 4.12 is not desirable at all. For releasing a deadlock, at least to vehicles are involved. The more vehicles are involved into a deadlock, the complicated it gets to solve the situation.

However, it does not only need time to resolve the deadlock, but also a massive amount of

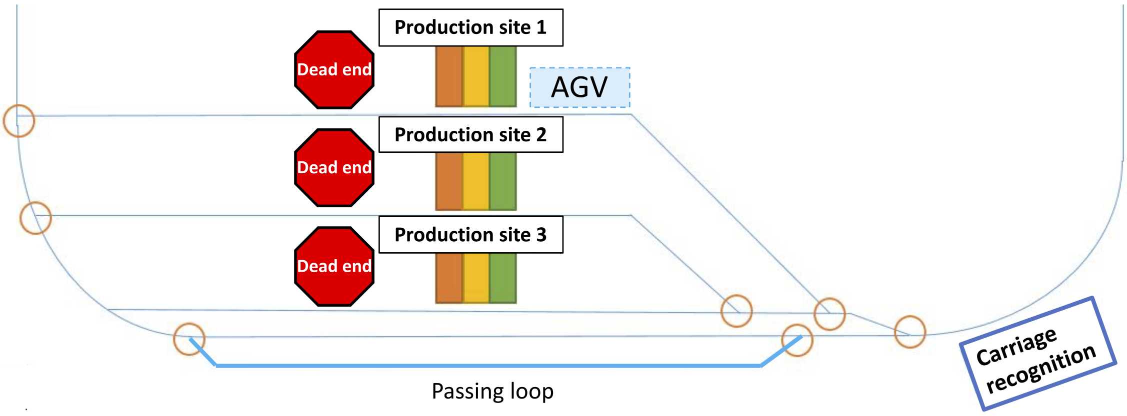


Figure 4.12.: Track excerpt for deadlock representation

coordination work, also from outside the system. The easiest way of deadlock prevention is to prevent letting two vehicles cross the paths. Only if the way is clear in front of the second vehicle, it shall proceed. As shown in figure 4.12, a minimum gap is required in order for the AGV to get out of the dead end, without having another vehicle moving. In the framework of a simple model, deadlock prevention shall be done by using the method described above. The blockchain as a distributed ledger however is not capable of preventing deadlocks in such systems.

4.5. Certification

Besides tracking material and product transaction history during the supply chain, blockchain technology can be used for different applications as well. An idea that came up during a personal interview with [Hin18], is the use of blockchain technology for certifying materials. During a supply chain process, many intermediaries are handling the goods. Here, a lot can be manipulated from farming the actual raw material until the finished product finds its way to the end consumer. Certainly, certification is important for every business division, but whenever humans have direct contact with the product, or rely on it, the situation gets critical. Examples for those kind of instances are the aviation industry, pharmaceutical products and food.

For deploying certificates into the blockchain for making them unchangeable, two steps are required. First, defining the parameters upon which the commodities shall be certified. Second, a method how to store those information and implementing the data carrier into the product itself. A project named "Everledger" [16] pursues the idea of tracking information about diamonds. The goal behind Everledger is to allow full transparency in product provenance. The core business model of Everledger is to make diamond provenance as transparent as possible so everyone can look at the history of the diamond. That way the potential buyer can assure that if the diamond was mined under legal conditions. This concept can be implemented into the manufacturing and logistics industry as well.

It is proposed, that the certifying starts even before farming the raw materials. In the following, an example is run through on wood as a raw material. The problem with wood is, that a lot of rainforests get deforested. In order to prevent that, every tree that is not meant to be chopped down, needs to be certified. In theory, every tree needs to be analysed an marked by The data carrier has to be inside the tree permanently. This is not a 100% tamper-proof, but it is sufficient enough, as the expense to find the implemented chip is higher than the actually reward that can be get from the tree. This is a basic principle for safety mechanisms. The effort has to be higher than the reward, that makes it obsolete for the attackers to try to crack the system.

[16] https://diamonds.everledger.io/

5. Conclusion

Throughout this paper a system for implementing the blockchain into a decentralised IoT enhanced industry environment has been proposed. Started with basic knowledge about the field of blockchain technology. This provides the reader with fundamental knowledge about an emerging technology. The motive for this is, that many people from outside the field cannot imagine much about the term *blockchain*. For this reason, the topic and the basic underlying mechanisms are presented within more detail here. In addition to the basics, extended technologies such as *Smart Contracts*, consensus rules and cryptography are introduced and presented. Furthermore, the topic of *Industry 4.0* is discussed. Basic concepts as well as the aspect of the *Internet of Things* are dealt with. The aspect of decentralised production control, as well as the implementation of automated guided vehicles are fundamentally presented due to the already broad existing knowledge.

In the main part of the work, the knowledge gained in the introductory part is conceptionally applied. For this, the vertical integration of a blockchain system is presented. A model train is addressed as a demonstration object, due to practical reasons. The feasibility of blockchain technology in an industrial environment is investigated. In addition to that, the integration of IoT devices is assumed, which should increase the degree of automation of the production environment even more. The interaction between IoT devices and the blockchain as data management, is also considered. For tracking purposes a custom coin, the *HAW Coin* is proposed upon the *Ethereum* blockchain. Here a scenario is played through, in which the blockchain interacts with automated guided vehicles, IoT devices as well as with decentralised autonomous organisations. The environment is initially considered with only one vehicle for the sake of simplicity. In the further course of the work, the scenario is presented in more realistic circumstances, by considering multiple vehicles in the environment. In the context of this observation, the topic of vehicle deadlock situations is briefly reviewed. In addition to the tracking aspect of the products during and after production, the approach of certification is also brought up. In this context, the provenance of the products is investigated. The focus is on critical products, which interact directly with people. This includes products from the aerospace industry, the food industry and many others.

6. Critical appraisal

Within the framework of scientific works, it is important to critically evaluate and scrutinise the acquired knowledge as well as the results obtained. This chapter is devoted to the critical examination of the results of this thesis.

6.1. Necessity of decentralisation

Before evaluating the role of the blockchain, the necessity of a generally decentralised production environment, with decentralised controlled vehicles needs to be clarified. This does not regard the use of blockchain technology yet.
As mentioned at the beginning of this work, there are advantages for centralised as well as decentralised systems. So far, centralised systems do work very well since the first industrial revolution and even before industrialisation at all. Decentralised systems have the big advantage of being very flexible and more independent towards other participants in the production system. In the future, the industry will surely grow rather than decline. Together with the industry, digitalisation will continue to increase. According to this fact, it is important to have systems working as reliable as possible. In a growing environment, a single-point of failure will have affect to more and more entities. As digitalisation will also grow in the industrial space, cyber attacks should always be considered. Within a centralised concept, one cyber attack can be enough to loot a big share confidential company data, as well as financial reserves. This would lead to major disruptions that have to be forfeited. The single-point of failure is one big problem centralised systems have to face, even more in greater company infrastructures. The aspect that decentralised structures lack a single-point of failure is very advantageous, particularly if one considers the continuing growth that will inevitably follow in the future.

6.2. Decison tree for reviewing blockchain use

The proposed blockchain concept is a prototype that has been presented for verifying the feasibility of industrial application. Blockchain technology got public attention very fast as crypto currencies experienced a vast hype in the year of 2017. Initiated by the rise of the *Bitcoin* price, all of a sudden everything needed a blockchain to be better [Rüe17]. Massive amounts of Initial Coin Offerings (ICO) got published. ICO's initiated by start ups that present projects mostly based on blockchain technology. For financing those projects, own launched coins are sold to the public. So before even introducing the concept on which the blockchain

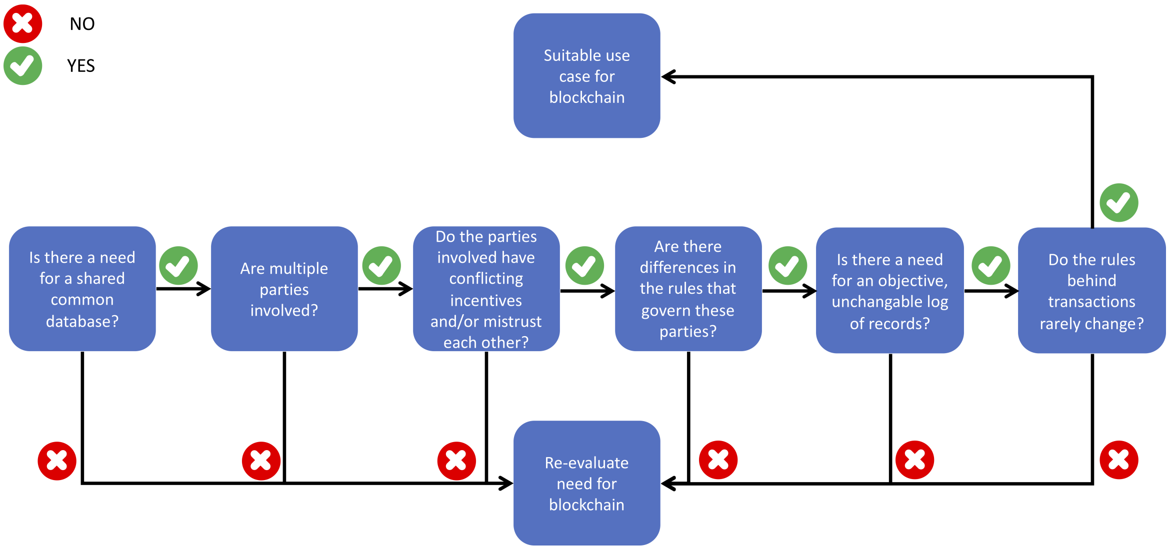


Figure 6.1.: Decision tree of blockchain necessity [MH18]

is applied to, it is of fundamental purpose to first check, if the usage of a blockchain in decentralised logistical environment even makes sense. Figure 6.1 shows a decision tree in which it can be determined, whether a blockchain application is useful or not. The illustrated decision tree by [MH18] is one of many models that enquire for the use of blockchain technology (see Figure 6.1). In the following the process is gone through step by step with regard to the concept examined in the framework of this elaboration.

Is there a need for a shared common database?

A shared common database is not imperative for general logistical purposes. Actually, in global and inter-company usage, it would be preferred to use unshared databases, due to confidential reasons. However for internal use, such as intra-logistical purposes common databases would make more sense. Here the aspect of confidentiality is not at risk. A common database means transparency on both sides and can lead to distortion of competition. The confidentiality of certain information always serves as a tool for strategic management. With a common database, many strategic business steps would no longer make sense. For having a common database companies need to be open to share their data. Partially common databases would be realistic for today's companies. Such as opening the database for one commodity for example. Therefore this question cannot be answered clearly but would tend more to "No".

Are multiple parties involved?

This question can be answered with a distinct "yes". This elaboration already deals with three different suppliers and three different manufacturing sites. This alone equals to six distinctive parties. In addition to that, multiple vehicles are also involved in the system. In a real world environment, the amount of parties and vehicles can be indefinite.

Do the parties involved have conflicting incentives and/or mistrust each other?

A certain amount of mistrust is always present in businesses. However, in established businesses, the business partners do not vary too much. Once having a few trustworthy partners, the business relationship with them will normally be maintained. On the other hand, a question arises why to do business with someone that is not trustworthy? Under today's circumstances, before doing business with an unknown party, the company is first analysed in depth. This counts even more, if the business generates more money.

Are there differences in the rules that govern these parties?

As long as acting within one and the same company, this can be excluded. In the case of business with other companies in different countries, it has to be reckoned, that the companies have different value concepts. This can be due to cultural or company variations. Additionally, working with international companies, law is also always a factor. Different countries are governed by different law.

Is there a need for an objective, unchangeable log of records?

An unchangeable log of records is beneficial for all business involved parties. Immutability is a form of security. Every company welcomes security, if the business is involved. A downside of immutability though is, that mistakes in the process can not be edited. Instead, the process has to be repeated from the beginning.

Do the rules behind transactions rarely change?

Rules behind the transactions in this consideration are implemented in the *Smart Contract*. As described throughout the paper, the *Smart Contract* is stored on the blockchain. This makes the *Smart Contract* unchangeable after deployment. If changes become necessary the *Smart Contract* needs to be shut down and a new contract must be drew up. Changes depend on the business department. Some businesses have a high frequency of changes, some not.

6.3. Critical review

As the results of the decision tree show, not all questions can be answered positively without any doubt. Neither by affirming or denying them. This provides the basis for a deeper entry and critical analysis into this topic.

The reason why the blockchain has attracted a lot of attention is, that in 2017 a large part of the crypto currencies recorded an exponential increase in value. Many of these crypto currencies, together with their own token, have also presented a potential use case. The blockchain technology was presented as the core technology for the marketing. Blockchain became the center of attention. This way, the blockchain got a real hype that nobody wanted to miss. However, behind none of these tokens there is a real product. In the context of this thesis, a lot of the companies which participated in ICOs, were analysed. Many of them are presenting ideas and plan on white-papers and road-maps. After all, none of these companies so far, has presented any of these proposed applications. It is very important to make clear, that blockchain technology alone is not a selling point. Like it was pointed out during this elaboration, that blockchain technology alone, is a relatively primitive data management solution.

A real application of the blockchain makes sense, if the trust model allows it. With regard to this work, the industrial aspect, especially the logistic area, is highlighted. Nowadays conventional companies are structured hierarchically. Broken down to the lowest tier, everybody always does the work for a person of higher authority (top-down). Thus, there are many points at which processes can be manipulated in favour of another party. As long as this is the case, a blockchain application is not appropriate, since decentralisation and the blockchain application presuppose equality of the interacting parties. The distributed consensus mechanism assures trust, without central authorities. The concept of DAO's was picked up in the context of this elaboration. This is a first step into the equality of positions within the companies, as well as beyond. Yet, this is very unlikely to happen at the present time. Every listed company strives to generate continuous growth. These companies are controlled by higher authorities such as shareholders. Those will not give up their position and financial benefits, in order to drive blockchain technology forward. Using crypto currencies on the basis of blockchain technology as a digital form of money is neither appropriate for business relations. In this case the currency *Ethereum* is proposed. Anyhow, crypto currencies in general are very volatile. That is why crypto currencies are still seen as objects of speculation. Even assuming that blockchain would be an appropriate tool for tracking purposes in industrial environment, the use of AGV presents the next challenge. The transport from A to B via an AGV, makes the vehicle a middleman. This is exactly the point that shall be prevented in blockchain application. This of course counts only in the consideration of external procurement. The implementation of a blockchain in such system would not be sufficient for the aforementioned reasons.

The blockchain application is more favourable, for certifying data. But here, too, there is the problem of implementation. Unfortunately, it is not possible with today's technology yet, to maintain the tagging of objects over the entire life cycle. Data carriers would have to be part of the material for the whole life cycle. Besides that the technology itself that should be implemented is a whole different challenge. An alternative would be to certify the end product only, with the history of the original composition of the product. However, here, too, one is faced with the challenge of correctly tracking the authenticity of the data without digital control from the extraction of raw materials to the finished product.

The aspect of certification is a field that should be investigated even more. Certifying especially critical and confidential products, such as in the aerospace industry, medical field or hazardous ingredients in food products. Certification could serve as a kind of indelible seal on products. Yet, this can be bypassed easily. The purpose for using the blockchain is absolutely given in ceritfying products, but the realisation without interfering external influence is not easily possible.

In conclusion it can be said that the blockchain technology is an emerging technology and needs time for further development and potential use cases. The point of decentralisation should be pursued further, as this represents the future in general automation in industry even without blockchain technology.

7. Guidance

Reviewing the achievements and findings in this work, it can be said that the implementation of blockchain technology is not suitable for tracking purposes in industrial environments, yet. However the subject of blockchain for industrial applications is still in its infancy. As with any other technology, it takes time to find mature applications to successfully implement it. Nevertheless, blockchain technology shall not fall into oblivion in the future. It is important to keep up with the latest developments in order to develop potential business ideas. Especially decentralisation in industrial environments will gain more and more attention in the future.

Future work

For driving the project of the decentralised model train environment forward, it is suggested not to implement blockchain technology as a distributed ledger for information and tracking purposes,yet. In addition to technical difficulties, during this work, the presented topic has not been found to be suitable for this purpose. For tracking the vehicle within the model, it is suggested to use passive RFID labels along the track. Simple location information can be stored on the RFID chips to use them as checkpoints. This easiest way would be to assign a number to each RFID tag. This way, an infinite number of tags can be integrated into the system. The interval of these RFID tags should equal along the whole track. This way it can also be used as a tool for speed measurement. The time between surpassing two checkpoint equals to the speed of the vehicle. In the framework of this project, the vehicle would transmit a response, every time it surpasses a checkpoint. This way, the vehicle can be located always between the range of two checkpoints. This is perfectly applicable with low effort in small scaled environments like the train model that has been presented in the course of this work. As the model grows, the RFID chips can also be used for routing purposes. Here fore, a sequence must be given, as a kind of track, that the vehicle has to follow. As long as the vehicle follows this route, one knows that it is not deviating from the given route. By sharing the route with the other AGVs, the route control can manage the traffic accordingly. The route control can be represented by an *Arduino* chip set. Every time the vehicle surpasses, it transmits a response.

To make sure, that the goods will arrive safely and in complete units at its destination, it is suggested to use more conventional methods. Every time an AGV is dispatched with goods, it gets a tracking number. By looking for this tracking number, the recipient can locate the goods according to the previously localisation method via RFID tags. Considering the speed of the train, which can be determined through the RFID tags as well, an estimated time of arrival can be calculated. This method is only as accurate as the distance of the RFID tags. Furthermore the RFID tags can serve as a decentralised communication protocol. Based on the location of a certain vehicle, this knowledge can be shared with other vehicles in the environment. This way, other AGVs know where not to operate. This can also be used for preventing deadlocks. However, geographically exactness is not needed in the most cases. For the most part it is not necessary to know the exact position of the items. It is more of an exception, that customers want to know where exactly their goods are all the time. In addition to this attitude, mistrust is created between the business partners. This also damages the business relationship.

Bibliography

[Bär14] Ralf Bär. Bär automotion gmbh, September 2014.

[BM16] Arshdeep Bahga and Vijay K. Madisetti. Blockchain platform for industrial internet of things. *Journal of Software Engineering and Applications*, 09(10):533–546, 2016.

[Bra17] Mathias Brandt. So viele jobs könnte die automatisierung kosten, March 2017.

[Bro10] Manfred Broy. Cyber-physical systems — wissenschaftliche herausforderungen bei der entwicklung. *Cyber-Physical Systems*, pages 17–31, 2010.

[But16] Vitalik Buterin. A proof of stake design philosophy. *medium.com*, December 2016.

[CDE+16] Kyle Croman, Christian Decker, Ittay Eyal, Adem Efe Gencer, Ari Juels, Ahmed Kosba, Andrew Miller, Prateek Saxena, Elaine Shi, Emin Gün Sirer, and et al. On scaling decentralized blockchains. *Lecture Notes in Computer Science*, pages 106–125, 2016.

[cyb15] Zukunft der arbeit in industrie 4.0. 2015.

[DH76] W. Diffie and M. Hellman. New directions in cryptography. *IEEE Transactions on Information Theory*, 22(6):644–654, Nov 1976.

[Don18] Dr. Franz-Joachim Kauffels; Andreas Donner. Die internet-protokolle im überblick – eine einführung, September 2018.

[Dre17] Daniel Drescher. *Blockchain Basics*. Apress, 2017.

[dub04] *Dubbel*. Springer Berlin Heidelberg, 2004.

[Fau18] Matthias Faul. Hochgenaue positionsbestimmung zur automatischen schiffsführung /. 08 2018.

[Flo18] David Floyd. The code for ethereum's concensus change is now ready for review, October 2018.

[Fra14] Pedro Franco. Understanding bitcoin. Oct 2014.

[Fur15] Prof. Dr. Kai Furmans. Industrie 4.0 und cyberphysische systeme: Kurzlebiger trend oder arbeitswelt von morgen? In *Fachtagung Warenlogistik*, 2015.

[Gar15] Jeff Garzik. Public versus private blockchains, part 1: Permissioned blockchains whitepaper. Whitepaper 1, Bitfury Group Limited, 2015.

[Her18a] Alyssa Hertig. Smart contracts for bitcoin? lightning's tadge dryja is working on it, October 2018.

[Her18b] Alyssa Hertig. What is a dao?, September 2018.

[Hin18] Dr. Johannes Hinckeldeyn. Interview about potential use cases for the blockchain technology, July 2018.

[Iva17] IvanOnTech. Building a blockchain in under 15 minutes - programmer explains, August 2017.

[Jah17] Myriam Jahn. Industrie 4.0 konkret. *essentials*, 2017.

[JM15] Gina Chung James Macaulay, Lauren Buckalew. Internet of things in logistics. Technical report, DHL Trend Research | Cisco Consulting Services, 2015.

[LA97] Peter Loos and Thomas Allweyer. Dezentrale planung und steuerung in der fertigung — quo vadis? *Organisationsstrukturen und Informationssysteme auf dem Prüfstand*, pages 83–99, 1997.

[Lab18a] Liezel Labios. These tags turn everyday objects into smart, connected devices, August 2018.

[Lab18b] Nest Labs. Starting page, August 2018.

[LAUP11] Q. Li, A.C. Adriaansen, J.T. Udding, and A.Y. Pogromsky. Design and control of automated guided vehicle systems: A case study. *IFAC Proceedings Volumes*, 44(1):13852–13857, Jan 2011.

[Lig18] Philips Lighting. Philips hue, August 2018.

[Lor18] Christian Lorenz. Wie von geisterhand, July 2018.

[LSP82] Leslie Lamport, Robert Shostak, and Marshall Pease. The byzantine generals problem. *ACM Transactions on Programming Languages and Systems*, 4(3):382–401, Jul 1982.

[Mar17] Martin. Industry 4.0: Definition, design principles, challenges, and the future of employment. *cleversim*, 2017.

[MH18] Dr. Markus Kückelhaus Matthias Heutger. Blockchain in logistics - perspective on the upcoming impact of blockchain technology and usecases for the logistics industry. *DHL Trend Research*, 2018.

[MHI12] MHI. *What are AGVs used for?* MHI, 8720 Red Oak Blvd., Suite 201, 2012.

[Mou16] William Mougayar. *The Business Blockchain: Promise, Practice, and Application of the Next Internet Technology*, volume 1. Wiley, 2016.

[Mün09] Technische Universität München. Proseminar kryptographische protokolle ss 2009 : Diffie – hellman schlüsselaustausch, June 2009.

[Nak08] Satoshi Nakamoto. Bitcoin: A peer-to-peer electronic cash system, 2008.

[OBNMBDHF06] Ludger Overmeyer, Dipl.-Ing Björn Niemann, Dipl.-Ing Mathias Baum, and Dr.-Ing Dirk-H. Fricke. Aufbau von fahrerlosen transportsystemen (fts) durch eine dezentrale datenstruktur. 11 2006.

[oT18] Ivan on Tech. Building a blockchain in under 15 minutes - programmer explains, February 2018.

[Pak18] Ingo Pakalski. Amazon enttarnt paketdiebe mit fingierten lieferungen, September 2018.

[Pau15] Sanjoy Paul. *Managing Disturbances in Supply Chain Systems*. PhD thesis, The University of New South Wales, 2015.

[Plu18] Werner Pluta. Die schauerleute von heute sind riesig und automatisch, August 2018.

[Rät18] HHLA/Thies Rätze. Hhla container terminal altenwerder, September 2018.

[Rau18] Raul. Transactions speeds: How do cryptocurrencies stack up to visa or paypal?, October 2018.

[Ros15] Christoph Roser. Industrial revolutions and future view, November 2015.

[Ros18] Meni Rosenfeld. How do bitcoin "colored coins" work?, Septembers 2018.

[Roß16] Peter Roßbach. Blockchain-technologien und ihre implikationen. *Frankfurt School of Finance Blog*, 2016.

[Rot16] Armin Roth. *Einführung und Umsetzung von Industrie 4.0*. Springer Berlin Heidelberg, 2016.

[Rüe17] Peter Rüegg. Blockchain ist ein hype, September 2017.

[Sau18] Martin Sauter. Schematic representation of vertical and horizontal integration in the automotive industry, October 2918.

[Sch06] Professor Dr.-Ing. Dipl.-Wirt. Ing. Günther Schuh. *Produktionsplanung und -steuerung*, volume 3. Springer-Verlag Berlin Heidelberg, 3 edition, 2006.

[Sch16] Vincent Schlatt. Blockchain: Grundlagen, anwendungen und potenziale. Technical report, Fraunhofer-Institut für Angewandte Informationstechnik FIT, 2016.

[sha17] sharda. An insight into the automated guided vehicle (agv) used in the maritime industry, October 2017.

[Sha18] Abishek Sharma. Understanding proof of stake through it's flaws. *medium.com*, January 2018.

[Six17] Elfriede Sixt. *Bitcoins und andere dezentrale Transaktionssysteme: Blockchain als Basis einer Kryptoökonomie*. Springer Gabler, Wiesbaden, 2017.

[SRI18] TAVISH SRIVASTAVA. Introduction to k-nearest neighbors: Simplified (with implementation in python), March 2018.

[SSLL14] Jochen Schlick, Peter Stephan, Matthias Loskyll, and Dennis Lappe. Industrie 4.0 in der praktischen anwendung. *Industrie 4.0 in Produktion, Automatisierung und Logistik*, pages 57–84, 2014.

[Swa15] Melanie Swan. *Blockchain: Blueprint for a New Economy*. O'Reilly Media, 2015.

[Sza18] Nick Szabo. Smart contracts, October 2018.

[Ull14] Günter Ullrich. *Fahrerlose Transportsysteme*, volume 2. Springer Fachmedien Wiesbaden, 2014.

[Uni17] Universa. Decentralized autonomous organization, 2017.

[Vel18] Lisa Velden. Der dijkstra-algorithmus, October 2018.

[Vos18] Dr. Shermin Voshmgir. What is a dao?, August 2018.

[Wen15] Thomas Wenzlaff. Arduino: Wie können rfid tags eingelesen und drahtlos empfangen werden?, Febriary 2015.

[Zep16] Tobias Zepf. Blockchain. technologien, innovationen und anwendungen. Master's thesis, Frankfurt School of Finance and Management, 2016.

[ZW16] Yu Zhang and Jiangtao Wen. The iot electric business model: Using blockchain technology for the internet of things. *Peer-to-Peer Networking and Applications*, 10(4):983–994, Apr 2016.

A. Blockchain

In this the core principle of the blockchain mechanism including the digital fingerprint is displayed in form of Java source code [Iva17]. First in A.1 the genesis block is created. In A.2 the concatenation of single block is represented.

A.1. Creating a block

Listing A.1: Creating the first block

```java
import java.util.Arrays;

public class Block {

        private int previousHash;
        private String[] transactions;

        private int blockHash;

        public Block(int previousHash, String[] transactions) {
                this.previousHash = previousHash;
                this.transactions = transactions;

                Object[] contens = {Arrays.hashCode(transactions),
                previousHash};
                this.blockHash = Arrays.hashCode(contens);
        }
        public int getPreviousHash() {
                return previousHash;
        }

        public String [] getTransaction() {
```

```
                return transactions;
        }
        public int getBlockHash() {
                return blockHash;
        }
}
```

A.2. Blockchain

Interlinking of the blocks to form a blockchain. [Iva17]

Listing A.2: Hashing of the transactions

```

import java.util.Arrays;

public class Block {

        private int previousHash;
        private String[] transactions;

        private int blockHash;

        public Block(int previousHash, String[] transactions) {
                this.previousHash = previousHash;
                this.transactions = transactions;

                Object[] contens = {Arrays.hashCode(transactions),
                 previousHash};
                this.blockHash = Arrays.hashCode(contens);
        }
        public int getPreviousHash() {
                return previousHash;
        }
```

```java

	public String [] getTransaction() {
		return transactions;
	}
	public int getBlockHash() {
		return blockHash;
	}
}
```

B. Own crypto currency

In section 4.3.2 an exerpt of the source code for launching the HAW Coin is presented. Listing B.1 represents the whole code.

B.1. Source Code for launching the HAW Coin

The settings for the custom coin are presented here in a prototypical manner. [oT18]

Listing B.1: Creating a custom coin based on the Ethereum blockchain

```
pragma solidity ^0.4.21;
import "./EIP20Interface.sol";
contract HAWCoin is EIP20Interface {

    uint256 constant private MAX_UINT256 = 2**256 - 1;
    mapping (address => uint256) public balances;
    mapping (address => mapping (address => uint256))
    public allowed;

    string public "HAWCoin";     //Name of the Coin to launch
    uint8 public 7;              //How many decimals to show.
    string public "HAW";         //An identifier

    function HAWCoin(
        uint256 _initialAmount,      //10000
        string _tokenName,           //HAWCoin
        uint8 _decimalUnits,         //seven
        string _tokenSymbol          //HAW
    )
    public {
        balances[msg.sender] = _initialAmount;
        // Give the creator all initial tokens
```

```
        totalSupply = _initialAmount;
        // Update total supply
        name = _tokenName;
        // Set the name for display purposes
        decimals = _decimalUnits;
        // Amount of decimals for display purposes
        symbol = _tokenSymbol;
        // Set the symbol for display purposes
    }
    function transfer(address _to, uint256 _value)
    public returns (bool success) {
        require(balances[msg.sender] >= _value);
        balances[msg.sender] -= _value;
        balances[_to] += _value;
        emit Transfer(msg.sender, _to, _value);
        return true;
    }
    function transferFrom(address _from, address _to,
    uint256 _value)
    public returns (bool success) {
        uint256 allowance = allowed[_from][msg.sender];
        require(balances[_from] >= _value && allowance >= _value);
        balances[_to] += _value;
        balances[_from] -= _value;
        if (allowance < MAX_UINT256) {
            allowed[_from][msg.sender] -= _value;
        }
        emit Transfer(_from, _to, _value);
        return true;
    }
    function balanceOf(address _owner) public view returns
    (uint256 balance) {
```

```
        return balances[_owner];
    }
    function approve(address _spender, uint256 _value)
    public returns (bool success) {
        allowed[msg.sender][_spender] = _value;
        emit Approval(msg.sender, _spender, _value);
        return true;
    }
    function allowance(address _owner, address _spender)
    public view returns (uint256 remaining) {
        return allowed[_owner][_spender];
    }
}
```

Erklärung zur selbstständigen Bearbeitung einer Abschlussarbeit

Gemäß der Allgemeinen Prüfungs- und Studienordnung ist zusammen mit der Abschlussarbeit eine schriftliche Erklärung abzugeben, in der der Studierende bestätigt, dass die Abschlussarbeit „– bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit [(§ 18 Abs. 1 APSO-TI-BM bzw. § 21 Abs. 1 APSO-INGI)] – ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt wurden. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich zu machen.“

Quelle: § 16 Abs. 5 APSO-TI-BM bzw. § 15 Abs. 6 APSO-INGI

Dieses Blatt, mit der folgenden Erklärung, ist nach Fertigstellung der Abschlussarbeit durch den Studierenden auszufüllen und jeweils mit Originalunterschrift als letztes Blatt in das Prüfungsexemplar der Abschlussarbeit einzubinden.
Eine unrichtig abgegebene Erklärung kann -auch nachträglich- zur Ungültigkeit des Studienabschlusses führen.

Erklärung zur selbstständigen Bearbeitung der Arbeit

Hiermit versichere ich,

Name: ______________________________

Vorname: ______________________________

dass ich die vorliegende – bzw. bei einer Gruppenarbeit die entsprechend gekennzeichneten Teile der Arbeit – mit dem Thema:

ohne fremde Hilfe selbständig verfasst und nur die angegebenen Quellen und Hilfsmittel benutzt habe. Wörtlich oder dem Sinn nach aus anderen Werken entnommene Stellen sind unter Angabe der Quellen kenntlich gemacht.

- die folgende Aussage ist bei Gruppenarbeiten auszufüllen und entfällt bei Einzelarbeiten -

Die Kennzeichnung der von mir erstellten und verantworteten Teile der ist erfolgt durch:

Ort	Datum	Unterschrift im Original