Process Automation on the Blockchain: An Exploratory Case Study on Smart Contracts

Julia Eggers
Technical University of Munich
julia.eggers@tum.de

Andreas Hein
Technical University of Munich
andreas.hein@tum.de

Jörg Weking
Technical University of Munich
joerg.weking@tum.de

Markus Böhm
Technical University of Munich
markus.boehm@tum.de

Helmut Krcmar
Technical University of Munich
helmut.krcmar@tum.de

Abstract

While business process automation through information technology has progressed over the last decades, smart contracts have recently emerged as a promising new means of automation. However, in practice, the adoption of smart contract-based automation is in its infancy, raising the question if the technology genuinely offers a unique approach to process automation. Drawing on an exploratory case study of four start-ups, we investigate the potentials for automation that organizations achieve through smart contracts and how smart contracts differ from established automation technologies, such as workflow management systems, enterprise resource planning systems, and robotic process automation. We contribute to the literature on process automation by unveiling transparent and immutable, cross-organizational, and decentralized automation as characteristics that differentiate smart contracts from established automation technologies. Besides, we provide practitioners with an understanding of application scenarios, potentials, and drawbacks of smart contracts for process automation.

1. Introduction

The human drive to eliminate hard manual labor is one of the principal reasons for the rapidly accelerating progress of automation technology. Automating value-creating processes has been ubiquitous during the evolution of humanity. Examples are the invention of the wheel, steam engine, and the computer. The computer significantly increased the transparency, reachability, and automatability of industrial as well as civil processes [1]. Instances of automation can be found everywhere in modern organizations, from an automatically brewed cup of coffee in the breakroom to complex procedures such as robots manufacturing products on an assembly line on the factory floor. However, despite significant technology-based advances in process automation, many firms are confronted with untapped automation potentials as they are struggling to create pervasive transparency and ensure the continuous, integrated, and automated execution of their business processes [2].

Under these circumstances, distributed ledger technology (DLT) and, specifically, smart contracts could hold the potential for the next revolutionary innovation toward extensive automation. DLT describes a distributed database or ledger wherein a group of peers share a copy and jointly scale, manage, and control operations [3]. On top of these ledgers, smart contracts are deployed to automate previously defined transactions if certain conditions in the contract's environment are fulfilled [3]. One prominent commercial application leveraging smart contracts for the supply chain is Everledger, which provides an immutable ledger to automatically track an asset's ownership and transaction history, such as in diamond trade. Instead of a third party manually recording a diamond's origin on paper or entering it into a local database—both methods prone to fraud—the source and transfer of ownership are automatically and permanently recorded on the blockchain using smart contracts. Because the blockchain is immutable, this renders an individual diamond's history not only transparent but also makes it traceable from the mine to the consumer, as well as prevents fraud and theft [4]. Smart contracts, therefore, might provide what established approaches to supply chain automation lack: pervasive, transparent, and distributed data handling for automating processes.

In sum, smart contracts promise a secure and transparent way to automate processes according to predefined rules and without the need to trust a third party to intermediate the transaction. Thus, a growing number of start-ups, for example, in the finance sector, are developing DLT use cases [5]. However,

URI: https://hdl.handle.net/10125/71301
978-0-9981331-4-0

according to a recent Gartner report, overall adoption is still in its infancy as incumbents are slow to implement the technology [6].

In light of their slow adoption rate, the question arises: do smart contracts genuinely offer a unique approach to process automation, or are they just the latest hype in the longstanding tradition of process automation technology? Practice demonstrates that most companies would rather trust in established automation techniques than adopt smart contracts. These techniques include, for example, enterprise resource planning (ERP) systems that are used by almost all small- to large-sized companies to integrate business functions, mirror process steps, and automatically execute specified process flows [1]. A second widely adopted substitute is the workflow management system (WfMS). WfMSs focus on controlling process operations concerning predefined operational sequences [7]. Intending to automate the time-consuming, individual steps of a process, robotic process automation (RPA) uses software bots that are programmed to imitate employee behavior by interacting with the front-end of information technology (IT) systems, processing tasks according to predefined rules [8].

Current research lacks a sufficient explanation for how the automation potential of smart contracts differs from that of well-established technologies [9]. Therefore, we are conducting this exploratory case study [10, 11] to unravel what automation potentials organizations might realize by applying smart contracts and if and how smart contracts differ from established automation technologies. Hence, we pose the following two research questions:

1) What automation potentials do organizations realize by applying smart contracts?

2) What are the differences between smart contracts and established automation technologies?

By addressing these two research questions, our exploratory study yields several implications for both research and practice. First, we contribute to the still limited theoretical understanding of the conceptualization and application of smart contracts for process automation. Our study reveals three distinct characteristics that differentiate smart contracts from established automation technologies while acknowledging the challenges that complicate the adoption of smart contracts in practice. Second, we provide practitioners with an understanding of application scenarios, potential benefits, and the drawbacks of smart contracts for process automation.

2. Theoretical Background

To contrast smart contracts with dominant process automation technologies, we first synthesize previous findings on DLT and smart contracts and present an overview of the advantages and disadvantages of conventional techniques for process automation, such as ERP systems, WfMSs, and RPA.

2.1. Distributed Ledger Technology

DLT is a type of database maintained by a peer-to-peer (P2P) network of nodes, which scale the database up and avoid the development of any single point of failure [12]. Peers can be distributed geographically as well as institutionally; moreover, each peer has a copy of the ledger [5]. The primary purpose of DLT is to enable untrusted peers to interact without the need for a trusted third party to mediate the interaction [12]. Therefore, a trusted ledger is based on two fundamentals. First, cryptographic means are applied to ensure immutable data on the ledger [5, 12]. Immutability is achieved through functionality such as digital signatures and fingerprints, taking the form of hash functions to ensure data validity. Second, peers agree on a particular state of the ledger to sustain a consistent data state [5, 12]. DLT, in general, encompasses a variety of implementations, the arguably most popular of which is the blockchain, which enables the realization of smart contracts [3].

2.2. Smart Contracts

Smart contracts are, in essence, programmed if-then-else conditions [13]. A smart contract can be a digital version of a contractual agreement between different parties or a relationship enforced by code, without any underlying contractual obligations or rights [3]. Even though the idea of the smart contract was already proposed by Szabo [14] in 1996, the realization of such a construct was not feasible at the time [13, 15]. Today, smart contracts are realized as decentralized, trusted, and shared code deployed on the blockchain, such as Ethereum or Hyperledger Fabric [16]. In this paper, we refer to the definition of smart contracts by Tapscott and Tapscott [15] as *"computer programs that secure, enforce, and execute settlement of recorded agreements between people and organizations"*. The computer program specifies the necessary predefined rules and ensures the automated execution of certain operations when corresponding trigger conditions are satisfied without the involvement of intermediaries [16]. The technological basis of smart contracts enables various potential use cases, ranging from product traceability along the supply chain to ensure regulatory compliance to defeating counterfeit products and fraud in industries such as healthcare, food, and energy [5].

One particularly promising application for smart contracts is the automation of business processes. Since smart contracts guarantee the automatic and transparent execution of predefined rules, they enable human judgment to be taken out of the equation [3]. An example is the automated handling of insurance policies deployed on the blockchain in the form of smart contracts. Once an insurance policy, such as a car insurance policy, is deployed as a smart contract, the smart contract automatically queries external sources for any events that might cause the filing of an insurance claim, for example, a crash that is detected by sensors in the car. In case such an event is detected and meets the predefined conditions, the smart contract automatically executes the terms of the insurance and issues the policyholder's payout. Therefore, the smart contract eliminates the need for trusted third parties and takes over the mediator role, directly connecting the parties in an inter-organizational context [17].

While smart contracts are promising for process automation, previous research has drawn attention to several challenges related to the understanding of and trust in the technology. First, for organizations to trust smart contracts, knowing how a specific smart contract is designed is necessary. However, it is difficult for people without technical expertise to read and therefore understand the source code of a smart contract [13]. Second, transparency as an essential DLT property also raises privacy concerns. For example, financial transactions are often seen as highly confidential and could, therefore, affect the adoption of smart contracts [18]. Third, the trustworthiness of the external data sources queried by smart contracts needs to be assured, for example, through a trusted third party [13]. Finally, the immutability of smart contracts is a challenging aspect for organizations, especially during the development and deployment phase, since errors cannot simply be erased with software updates [16].

2.3. Enterprise Resource Planning Systems

While organizations rely on various IT systems to support their business processes, such as customer relationship management or supply chain management systems, ERP systems arguably emerged as one of the most prominent and widespread IT systems used in almost any organization to manage resources and support business processes [1]. Thus, our analysis exemplarily focuses on ERP systems. Due to the widespread use and improvement of ERP systems over the years, they now offer pre-built, best practice-based automation options to automate standard processes such as accounting or order procedures. Adopting an ERP system thus offers an organization the opportunity to transfer tacit knowledge on process automation. One example is the University of Nebraska, which implemented an ERP system to improve accounting, inventory, and purchasing processes. After implementing the new system, a standardized, rule-based, centrally controlled procedure offered inherently by the system design was adopted for purchasing processes [1]. However, inheriting system-induced best practices and transferring tacit knowledge on process automation requires the adopting organization to adapt or omit existing processes significantly. Such enforced adoption of new, partly automated processes might stir conflict in the workforce that can be time-consuming and difficult to resolve [19]. In addition, ERP adoption and customization are usually associated with substantial implementation costs [1].

ERP systems not only provide pre-built automation solutions but also offer the data foundation necessary to improve existing automation. As data storage in a single location requires the enforcement of strict database rules for all data, including type, form, and accessibility, analytics technology can be deployed upon the vast amount of standardized data available in ERP systems. For example, analytics algorithms based on pattern recognition can automatically switch running processes to alternative routines in case of an erroneous process execution [20]. At the technical level, however, the administration and management of data from various sources require investment in the infrastructure and resources required to transform data into valuable insight [20].

2.4. Workflow Management Systems

Workflow management describes the control of document operations in terms of roles and responsibilities, associated access rights, and the operational sequence of document editing [21]. In general, WfMSs bear great potential for rule-based, low-level process automation. Based on predefined workflows modeled by the user, the system automatically ensures document consistency and availability within a process, automatically triggers sub-process parallelization if suitable, and automatically executes the steps of the process following predefined business rules [21]. In addition, WfMSs are extendable to interact with sub-systems or other WfMSs across company borders through web services or middleware, which is a key factor for automating distributed business processes [19, 22]. In recent years, WfMSs have been augmented with distributed data tracking functionality, such as radio frequency identification (RFID) sensors, which allow, for example, the granular tracking of items in the manufacturing workflow. This approach is similar to smart contract-enabled distributed data tracking [23].

However, WfMSs require continuous time- and resource-consuming analyses to ensure their effectiveness. Such analyses include testing to ensure the predictability of workflow behavior, testing to eliminate redundancies and deadlocks, and data usage analysis to prevent fraud [24]. Accordingly, dynamic changes in workflows at runtime pose a significant challenge as they contradict system predictability and controllability [21]. Besides, reflecting actual processes in the WfMS can be problematic since various process variants usually coexist. Because WfMSs are not designed to represent this complexity, an oversimplification of the organizational process dimension results in either WfMS inflexibility or exceeded liberality, thereby deteriorating the effectiveness of the system [7]. Finally, associating large amounts of data, for example from RFID, potentially increases the complexity of integrating WfMSs with other enterprise information systems, thus increasing the amount of time required to develop, test, and maintain process automation [23].

2.5. Robotic Process Automation

RPA has gained momentum in recent years as a lightweight, easy-to-configure IT solution that enables organizations to automate business processes [8]. To achieve process automation, RPA leverages software bots that imitate human-system interactions [25] without requiring the underlying system infrastructure or code to be changed [8]. An RPA bot is designed to access the system through the user interface, similar to how an employee would use an IT system. Then, the bot processes the tasks according to predefined rules and delivers the data to subsequent systems [26]. Tasks that are particularly suitable for automation using RPA are often referred to as "swivel chair" processes [8], for example, when an employee takes electronic input from one system, processes it according to predefined rules, and then enters the data into another system. RPA aims to relieve employees of monotonous tasks, thereby increasing their job satisfaction as they become free to focus on more complex and creative tasks [8]. In addition, overall productivity and product and service quality increase due to the faster and less error-prone task execution of bots [26]. Although RPA bears significant automation potential, its economically valuable application remains limited to rule-based processes that are time-consuming yet unlikely to yield unexpected events [25]. The strong dependency on predefined patterns excludes the application of RPA for tasks that require common sense, a creative approach, or exception handling [8]. Further technical limitations are due to the underlying IT systems that are designed for human utilization. Thus, they may not be able to cope with bot speed, leading to errors and process failures [8].

While there are various technologies available for realizing process automation, it should be noted that scholars have long debated whether an increase in process performance is not only able to be achieved by automating processes through technologies such as ERP systems, WfMSs, or RPA bots but also by radically reengineering or obliterating processes [27]. The basic assumption is that the automation of ineffective or cumbersome processes might result in rather undesirable outcomes, such as slow, erroneous business processes. While we agree with the importance of comprehensive business process management and reengineering as an antecedent to process automation, we focus our study on the implementation phase of process automation as it is enabled by different technologies.

3. Methodology

Sampling Strategy. Due to the recent emergence of smart contracts as a field of research and their scarce application in practice, we chose to conduct an exploratory case study [10] drawing on various cases. The case study design allowed us to study the contemporary phenomenon of the smart contract as an automation tool in its organizational context. Furthermore, the analysis of multiple cases ensured robust findings using literal replication [10]. We selected four organizations that met the following criteria: (1) offering a product based on smart contracts and (2) using smart-contract technology to automate part of the value generation process. The four organizations chosen were operating as start-ups in the real estate, insurance, logistics, and security industry, respectively (see Table 1).

The first case is *ImmoCorp*, a Switzerland-based start-up founded in 2018 operating in the real estate market. *ImmoCorp* uses smart contract-based technology to create tokenized, digital twins of properties, meaning that buildings are represented by a defined number of digital tokens that could then be bought or sold through the *ImmoCorp* platform. Properties are sold in "crowdsales," referring to multiple buyers jointly acquiring the real estate. After the acquisition, owners can use the platform to vote on matters concerning their property.

Second, we chose *InsurCorp*, a German start-up founded in 2016 that offers a smart contract-based platform for decentralized insurance products. For our study, their parametric flight delay insurance was of particular interest since this application enables the automated and decentralized pricing, underwriting, and processing of insurance policies for flight delays.

The third case is *LogiCorp*, a Switzerland-based start-up founded in 2016 that uses smart-contract technology in combination with sensors to automatically monitor the end-to-end shipment of products prone to strict temperature regulation.

Fourth, we included the start-up *SecurCorp*, founded in 2017 in the US that offers a decentralized marketplace for threat intelligence using smart contracts. The marketplace automatically connects anti-malware providers with end customers to provide threat detection products aligned with their specific needs.

Data Collection. To ensure construct validity [10], we used multiple sources of evidence, including semi-structured expert interviews [28] and archival data, such as case studies, product demos, and newspaper articles. Our interview questions focused on three areas of inquiry: how the firm uses smart contracts, why they use smart contracts to automate part of the value generation process, and how smart contracts differ from alternative automation techniques available. To ensure reliable results, we maintained a chain of evidence through a case database [10].

Data Analysis. We engaged in explanation building [10] as an analytic technique, wherein we continuously iterated between initial explanatory propositions and the findings from the cases to explain why organizations chose smart contracts as automation technology. To provide for external validity, we conducted a cross-case analysis to demonstrate the transferability of results from the within-case analysis to a context outside of the specific case scenario [11].

Table 1. Overview of case studies

Company	Industry	Interviews	Archival	Demo
ImmoCorp	Real Estate	2	5	Yes
InsurCorp	Insurance	1	5	Yes
LogiCorp	Logistics	1	6	No
SecurCorp	Security	2	5	Yes

4. Results

Our study reveals automation by disintermediation and automation by reducing manual process steps as key automation mechanisms by smart contracts.

4.1. Automation by Disintermediation

First, smart contracts enable firms to design and automate processes by skipping intermediaries. In the case of *InsurCorp*, applying smart contracts allowed the organization to fundamentally change insurance processes. As in the case of flight delay insurance, insurance services usually rely on a complex process involving multiple actors, such as insurance brokers, who negotiate the policy with potential policyholders, and the underwriter that manages the cash flows. *InsurCorp*, however, uses smart contracts to skip brokers and even the insurance company, so the customer only interacts with the smart contract. The insurance policy and all related assets and processes, including pricing, underwriting, issuing, and claim settling, are now deployed as a smart contract with DLT. Triggered by external events, such as the delay of an insured flight, a smart contract automatically executes predefined steps. In the event of a claim, the smart contract automatically pays out the insurance sum to the policyholder, skipping the intermediating insurer, as an *InsurCorp* employee described:

"[Customers] do not have to ask anyone about the money, they are not dependent on an insurance company whether this payout happens or not, but it happens automatically, solely based on data. [...] There is no middleman anymore; no one is mediating between the insured parties and gets paid to manage these cash flows."

However, due to strict regulations in specific industries, such as the real estate market, it is currently impossible to eliminate all intermediaries with smart contracts. In the case of *ImmoCorp*, the platform must verify each new user using third-party identification services. In addition, each "crowdsale" of a tokenized property on the *ImmoCorp* platform requires an off-chain notary to approve the sale. Nevertheless, the smart contract-based platform eliminates the need for a bank to act as an intermediary to facilitate the transaction and manage cashflows. A user is only required to own a wallet that represents a public and private key used to receive or spend cryptocurrency on the blockchain. When properties on the platform are sold in "crowdsales," the smart contract ensures that all payments are automatically collected and stored on the blockchain. After a predetermined sales period expires, the "crowdsale" is closed, and, in case the desired price has been achieved, an off-chain notary approves the sale. Next, the smart contract pushes the automatic transfer of the sum paid to the seller(s), while ownership of the property automatically changes. Therefore, users and the platform become independent of banks and real estate agents, providing them with more flexibility in buying and selling tokenized properties, as described by a technology expert from *ImmoCorp:*

"You can sell [the tokens of a property] again the next day if you want to, and you have everything under control yourself. You do not need middlemen;

you do not have to see a bank advisor or a lawyer; you can do it yourself from home."

By eliminating the bank as an intermediary, the payment procedure for a property is automated from end to end. Instead of issuing a wire transfer that requires one bank to allocate the money from the buyer's bank account to the seller's account—sometimes prompting the conversion of currencies—the money is directly and automatically transferred via the smart contract.

Drawing on smart contracts to automate end-to-end cross-organizational processes enables *SecurCorp* to dissolve formerly rigid company boundaries. *SecurCorp* leverages smart contracts to automate transactions in the global anti-malware marketplace using a common cryptocurrency as the means of payment. Based on the Ethereum blockchain, the smart contract-based application provides a shared infrastructure for customers and providers from the anti-malware industry to interact directly and automatically exchange documents, assets, and information across company borders. Therefore, the automation of transactions becomes independent of the customer's and provider's individual local infrastructures but relies entirely on the smart contract to function as a shared infrastructure that can support an end-to-end transaction. Consequently, the payment procedure is automated through smart contracts. Previously, customers manually paid the provider in their local currency via wire transfer, possibly requiring a currency conversion by an intermediating bank. Now, the smart contract can automatically transfer cryptographic assets between the customer and provider according to predefined rules, as an employee of *SecurCorp* explained:

"*We want to be a worldwide marketplace. Furthermore, if you are not using some form of a fixed token, then you have to deal with fiat in every country where people want to participate. So instead of us dealing with fiat, we are dealing with a single token [and so] we need a smart contract.*"

4.2. Automation by Reducing Manual Process Steps

Smart contracts provide new means for automated and secure data tracking and distribution, thereby automating process steps that otherwise need to be performed manually. *LogiCorp* uses these properties to automatically and reliably track the supply chain of temperature-sensitive pharmaceuticals. Regulations require that temperature-sensitive products be monitored during delivery and that temperature data be distributed to the vendor and buyer upon shipment to ensure regulatory compliance. Formerly, tracking and distributing temperature data was a highly labor-intensive process involving the manual programming of temperature sensors and the manual collection, analysis, and redistribution of data from individual sensors. With smart contracts, each shipment persists on the blockchain, containing the predefined permissible temperature range and an identifier for the temperature sensor placed in the shipment. This process ensures that the logged temperature data is immutably stored on the blockchain. Upon delivery, the sensor is scanned, which triggers the smart contract to automatically analyze the data and send a report to all involved stakeholders. Smart contracts enable faster automated monitoring of the supply chain, as a founder of *LogiCorp* explained:

"*It was a cumbersome procedure, and it relied on much work between sender and receiver and came with time delays. But now being able to know [the temperature data] immediately upon arrival prevents shelving of compromised goods and keeps senders automatically aware of any challenges they are having in the supply chain.*"

SecurCorp leverages smart contracts to automatically execute predefined actions that enable automated matching of service providers and customers in the anti-malware market. Formerly, customers either relied on per-seat license models to acquire a single anti-malware package from incumbent software providers or directly approached specialized anti-malware providers for specific problems. By contrast, the *SecurCorp* platform creates a marketplace in which customers are able to submit a digital artifact to be checked for malware by different service providers. The smart contract-based platform then automatically distributes the request to anti-malware service providers, according to their predefined capabilities, that are registered on the platform. These service providers offer specialized software that automatically first analyzes the received artifact and then replies to the customer with a verdict (i.e., whether the artifact is malicious or benign) and an assessment regarding their verdict's certainty. Also, a synthesized recommendation based on a majority vote is sent to the customer that is recorded on the blockchain, allowing transparency on how anti-malware providers perform over time. Therefore, the smart contract automatically facilitates the matching of customer and provider so that customers no longer need to search for one suitable security expert manually. Instead, they automatically receive an assessment from multiple experts, as described by a *SecurCorp* employee:

"*[T]he smart contracts are doing all of the transactions between the people submitting the files* *and the people analyzing the files. Moreover, [the smart contract is] recording the results. So, the function of the smart contract is really to allow the people who have a file to submit it automatedly*."

5. Discussion and Conclusion

There are three key differences when comparing smart contract-based process automation with established automation technologies, such as ERP systems, WfMSs, and RPA.

5.1. Transparent and Immutable Automation

First, we observe that smart contracts offer a transparent and immutable manner for automating processes due to their underlying blockchain technology. While immutability on the blockchain is not new, this combination with automation enables companies to automate business processes according to predefined, unchangeable rules. As in the case of *InsurCorp*, the rules for automated insurance payout are specified in the smart contract available on the Ethereum blockchain. Therefore, the insurance policy is fully transparent to the customer. The smart contract and the subsequent insurance payout, which cannot be changed once signed, are publicly available and accessible to everyone. The transparency and guaranteed immutability of automation rules are likely to increase users' trust in smart contract-based automation. Research has long shown that transparent process automation, referring to users knowing the inner logic and rules of the automated system, fosters a higher level of user trust and human-automation task performance [29]. Building trust in financial transactions through increased transparency could be one reason for organizations to apply smart contracts in the future, for example, to automate trade financing. There are examples in practice already, such as the case of a large Australian bank that advanced itself as one of the first financial institutions to apply smart contracts to automatically and transparently process invoice financing documents [30].

On the contrary, for established process automation technologies like WfMSs, RPA, and ERP systems, there is no guarantee of such transparency. Automation rules might not be clear and accessible to all users of the system. Instead, they are known only by system administrators or programmers [31], leaving most system users unaware of the underlying automation procedures. This effect intensifies as the automation rules in WfMSs, ERP systems, and RPA bots are modifiable and, thus, can be altered without all users being aware of the changes [31]. However, the option of changing process automation declarations after being deployed leaves room to adapt to changing environmental conditions, such as new regulatory requirements or varying customer expectations [21].

In this regard, smart contracts are limited in their ability to adapt to shifting process requirements. Because immutability is a core concept of the blockchain, once a smart contract is deployed on the blockchain, it cannot be changed. Instead, desired changes must be deployed as a new smart contract, which then needs to be referenced by all stakeholders while the previous smart contract is simultaneously disabled [16]. This complex updating procedure results in increased tension arising between inflexible process automation and quickly changing business conditions. We observed this complication at *ImmoCorp*, which is subject to the changing regulations of the real estate industry. Therefore, the firm was required to develop a complex proxy-based architecture workaround to enable frequent changes in the smart contract's automation rules.

In conclusion, compared to established automation technologies, smart contracts offer a more transparent means for process automation. The underlying logic is guaranteed to be both reliable and accessible to all users. This increased transparency is likely to increase user trust in automated transactions. As a downside, automation with smart contracts suffers from rule rigidity and the inflexibility to adapt to changing requirements.

5.2. Cross-Organizational Automation

Second, organizations achieve end-to-end automation of cross-organizational processes using smart contracts. As observed in the case of *SecurCorp*, smart contracts provide the necessary company-spanning, decentralized infrastructure that is required to connect customers' and providers' local infrastructures. A centralized authority does not maintain the infrastructure. Therefore, several organizations can cooperate without the need to agree on a third-party provider [17]. Building on the blockchain, cross-organizational automated transactions such as document and asset exchange become feasible through the intermediating actions specified in the smart contract. In addition, the automation rules are transparent to all involved parties, which can promote increased trust in the automated process itself. These findings are consistent with a previous study that demonstrated that a blockchain-based system could facilitate cross-organizational process management and automation by providing a common and transparent infrastructure [17]. Therefore, one reason why organizations choose smart contracts might be to automate processes across different systems and company boundaries transparently. One promising area of application is the healthcare industry, which is today characterized by many actors and systems being necessary to execute healthcare services, such as filling a prescription. Smart contracts are expected to provide the required, low-cost infrastructure to automate transactions between healthcare providers, health insurance companies, and patients [32].

Still, cross-organizational process automation is considered a complex endeavor, and there is consensus that not all automation technologies are suitable. RPA, for example, is not considered to be a technological fit for cross-organizational end-to-end process automation. Instead, RPA is designed as a tool operating on the user interface of IT systems with the goal of intra-organizational, low-level automation aimed at individual steps of a process [26]. ERPs and WfMSs, however, can be applied by organizations to realize cross-organizational process automation. Although used at the individual enterprise level, these systems can be integrated between organizations at the infrastructure level, drawing on, for example, middleware or web service technologies to automatically exchange data across company boundaries [19]. However, implementing the solutions raises technological and governmental challenges such as agreeing on shared infrastructure providers [17], ensuring data flow transparency, and deciding on responsibility for data entry and updates [19]. In addition, concerns regarding data privacy and security become essential [33]. Connecting ERPs or WfMSs between organizations bears the risk of unintentionally sharing sensitive data, such as employee- or finance-related information, with another organization [22]. Therefore, firms perform complex data anonymization or filtering procedures [33] before sharing data.

Data privacy also plays a significant role in smart contract-based process automation. The example of *ImmoCorp* indicates that the procedural transparency offered by smart contracts poses the risk of storing and sharing sensitive transactional data such as the name and address of real estate investors with all parties that have access to the blockchain. *ImmoCorp* is, therefore, relying on a hybrid approach in which sensitive information is stored in off-chain databases, and only the storage address persists on the blockchain. These findings support previous work arguing that the lack of data privacy is a drawback of smart contacts [18].

In conclusion, companies realize cross-organizational end-to-end process automation using smart contracts as a shared automation infrastructure that connects organizations' local IT systems according to transparent rules. Similarly, cross-organizational process automation can be achieved by integrating WfMSs or ERP systems between company boundaries employing web services or middleware systems. However, these types of automation are not necessarily transparent to all stakeholders and often depend on trusting a third-party infrastructure provider. Besides, there is the risk of unintentionally sharing sensitive data between companies through end-to-end process automation, independent of the chosen technology.

5.3. Decentralized Automation

Lastly, organizations leverage smart contracts to enable decentralized process automation independent of intermediaries or third-party providers. Disintermediation is a frequently cited advantage of smart contracts [16]. In the context of process automation, it allows for faster process execution by automating or skipping process steps that were formerly related to intermediaries. The example of *InsurCorp* demonstrates how smart contracts eliminate the need for an insurance company to act as an intermediary since the insurance policy and all related coordinating processes are deployed as a smart contract on the blockchain. Therefore, both customer-facing processes, such as insurance payouts, and back-office processes, such as flight delay checks, are performed automatically and without a third-party coordinating or executing the transaction. For this reason, *InsurCorp* claims to provide a faster payout process than incumbent insurance companies.

In addition, neither stakeholders nor third parties need to host the shared infrastructure since the blockchain is designed as a decentralized P2P network. The decentralized infrastructure also ensures that the stakeholders no longer need to trust a potentially unknown third-party provider who controls the automation infrastructure. Instead, control of the blockchain-based infrastructure is distributed among all transaction stakeholders. In contrast, WfMSs, RPA, or ERP systems provide a centralized approach to automation that usually requires the stakeholders to trust a third party, such as consultants, vendors, or IT departments, to build and maintain the corresponding infrastructure [34]. Especially in the insurance industry, smart contracts appear to be a promising technology for not only decentralizing and automating processes but also facilitating new business models independent of intermediary organizations. Today, this trend can be observed in a growing number of start-ups offering smart contract-based insurance products, ranging from maritime freight insurance to pet or unemployment insurance [35]. At the same time, incumbent insurers face the challenge of adopting smart contracts that inherently pose the risk of rendering them obsolete by taking over the insurance company role [36].

However, successfully leveraging smart contracts for decentralized automation depends on the collaboration of all stakeholders, including customers, vendors, and logistics providers. Persuading stakeholders to use smart contracts can be challenging because doubts and misconceptions about the new technology prevail. *LogiCorp* was challenged in convincing supply chain stakeholders, such as pharmaceutical producers and logistics providers, to engage in supply chain process automation using smart contracts. Stakeholders were insecure about the technical functionality and data security, and only after *LogiCorp* delivered proof of concept on the implementation of smart contracts was a consensus reached. Similar concerns can also be currently observed in the emerging RPA market, as companies are still skeptical about the functionality and benefits of RPA-based process automation [26]. In contrast, established automation technologies with a high market share, such as WfMSs and ERP systems implemented by well-reputed vendors, are perceived as rather trustworthy and useful [34].

Overall, organizations choose smart contracts to achieve efficient, decentralized process automation that is no longer dependent on a centralized infrastructure or requires trust in intermediaries. However, contrary to established automation technologies, such as ERP systems and WfMSs, firms still struggle to convince process stakeholders to adapt smart contracts-based automation due to the technology's novelty.

5.4. Limitations and Conclusion

Even though smart contracts promise a secure and efficient way to automate business processes, in practice, the adoption rate is rather slow [9]. Therefore, our study set out to explore what automation potentials organizations might realize by applying smart contracts and how smart contracts differ from established automation technologies. By drawing on four case studies, we demonstrated that organizations implement smart contracts to achieve transparent, immutable, cross-organizational, decentralized process automation.

However, our study is subject to several limitations. First, by studying only organizations that have implemented smart contracts to achieve process automation, our study does not cover the perspective of firms that have rejected smart contracts and opted for alternative automation technologies. While our study provides a primary exploration of the characteristics that make smart contracts a valuable automation solution, we suggest that future research investigate the challenges of implementing smart contracts along with reasons for choosing alternative automation technologies. Second, our research was designed as a highly exploratory endeavor due to the novelty of the technology and the sparse extant research on its application. Therefore, we investigated four start-up companies from a variety of industries taking a qualitative approach, which provided insights regarding the still unexplored reasons that firms automate processes using smart contracts. Still, we encourage scholars to systematically further knowledge on the benefits and challenges of smart-contract applications for process automation, specifically for incumbent firms that usually maintain legacy systems and complex processes.

In light of the low adoption rate of smart contracts in practice, especially for incumbent firms, the question arises if smart contracts are only the newest hype in a longstanding tradition of process automation technologies. Our study shows, however, that smart contracts offer distinct automation opportunities compared to established automation technologies. Smart contract-driven automation enables organizations to eliminate formerly required intermediaries while increasing stakeholder trust in automation through transparent and unchangeable automation rules recorded in the smart contract. In addition, the deployment of smart contracts on the blockchain provides a company-spanning automation infrastructure that is independent of a third-party provider—in contrast to ERP, WfMSs, or RPA providers—but instead jointly maintained by all process stakeholders. However, smart contracts should not be considered a universal approach to automation. Today, their implementation is challenged by privacy concerns due to the complete transparency of the blockchain, by inflexibility due to the blockchain's immutability, and by the skepticism of stakeholders who are slow to adopt the new technology.

6. References

[1] Lee, Z. and J. Lee, "An ERP Implementation Case Study From a Knowledge Transfer Perspective". Journal of Information Technology, 2000. 15: p. 281-288.

[2] Manyika, J., M. Chui, and M. Miremadi. "A Future that Works: AI, Automation, Employment, and Productivity", McKinsey Global Institute, 2017. Available from: https://mck.co/35AmmkK.

[3] Swan, M., "Blockchain: Blueprint for a New Economy". 2015, Sebastopol, California, USA: O'Reilly Media.

[4] Everledger. "Everledger. Our Technologies", 2020 [cited 02.07.2020]. Available from: https://www.everledger.io/our-technologies/

[5] Weking, J., et al., "The Impact of Blockchain Technology on Business Models–a Taxonomy and Archetypal Patterns". Electronic Markets, 2019. 30: p. 285–305.

[6] Clougherty Jones, L., et al. "Predicts 2020: Data and Analytics Strategies — Invest, Influence and Impact", Gartner, Inc., 2019. Available from: https://gtnr.it/3kxSHwM.

[7] Kumar, A., W.M. Van Der Aalst, and E.M. Verbeek, "Dynamic Work Distribution in Workflow Management Systems: How to Balance Quality and Performance". Journal of Management Information Systems, 2002. 18(3): p. 157-193.

[8] Lacity, M. and L.P. Willcocks, "Robotic Process Automation at Telefonica O2". MIS Quarterly Executive, 2016. 15(1): p. 21-35.

[9] Cai, X., X. Zhao, and B. Zhang. "Identify Multiple Types of Social Influences on Smart Contract Adoption in Blockchain User Network: An Empirical Examination of CryptoKitties in Ethereum", in Proceedings of the International Conference on Information Systems (ICIS). 2019. Munich, Germany.

[10] Yin, R.K., "Case Study Research: Design and Methods". 4 ed. Applied Social Research Methods, ed. L. Bickman and B.J. Rog. 2009, Thousand Oaks, CA: Sage.

[11] Eisenhardt, K.M., "Building theories from case study research". Academy of Management Review, 1989. 14(4): p. 532-550.

[12] El Ioini, N. and C. Pahl, "A Review of Distributed Ledger Technologies", in On the Move to Meaningful Internet Systems. OTM 2018 Conferences. OTM 2018., Panetto H., et al., Editors. 2018, Springer: Cham.

[13] Cuccuru, P., "Beyond Bitcoin: An Early Overview on Smart Contracts". International Journal of Law and Information Technology, 2017. 25(3): p. 179-195.

[14] Szabo, N., "Smart Contracts: Building Blocks for Digital Markets". EXTROPY: The Journal of Transhumanist Thought, 1996. 18(2).

[15] Tapscott, D. and A. Tapscott, "Blockchain Revolution: How the Technology Behind Bitcoin is Changing Money, Business, and the World". 2016, New York, USA: Portfolio/Penguin.

[16] Wang, S., et al. "An Overview of Smart Contract: Architecture, Applications, and Future Trends", in Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV). 2018. Changshu: IEEE.

[17] Fridgen, G., et al. "Cross-Organizational Workflow Management Using Blockchain Technology - Towards Applicability, Auditability, and Automation", in Proceedings of the 51st Hawaii International Conference on System Sciences. 2018. Maui, Hawaii.

[18] Kosba, A., et al. "Hawk: The Blockchain Model of Cryptography and Privacy-Preserving Smart Contracts", in Proceedings of the 2016 IEEE Symposium on Security and Privacy (SP). 2016. San Jose, CA.

[19] Daneva, M. and R. Wieringa. "Requirements Engineering for Cross-Organizational ERP Implementation: Undocumented Assumptions and Potential Mismatches", in Proceedings of the 13th IEEE International Conference on Requirements Engineering (RE'05). 2005. Paris, France: IEEE.

[20] Mikalef, P., et al., "Big Data Analytics Capabilities and Innovation: The Mediating Role of Dynamic Capabilities and Moderating Effect of the Environment". British Journal of Management, 2019. 30(2): p. 272-298.

[21] Kumar, A. and J.L. Zhao, "Dynamic Routing and Operational Controls in Workflow Management Systems". Management Science, 1999. 45(2): p. 253-272.

[22] Poulymenopoulou, M., F. Malamateniou, and G. Vassilacopoulos, "Emergency Healthcare Process Automation Using Workflow Technology and Web Services". Medical Informatics and the Internet in Medicine, 2003. 28(3): p. 195-207.

[23] Cheng, C.Y. and V. Prabhu, "An Approach for Research and Training in Enterprise Information System with RFID Technology". Journal of Intelligent Manufacturing, 2013. 24(3): p. 527-540.

[24] Basu, A. and A. Kumar, "Research Commentary: Workflow Management Issues in E-Business". Information Systems Research, 2002. 13(1): p. 1-14.

[25] Hallikainen, P., R. Bekkhus, and S.L. Pan, "How OpusCapita Used Internal RPA Capabilities to Offer Services to Clients". MIS Quarterly Executive, 2018. 17(1): p. 41-52.

[26] Willcocks, L.P., M. Lacity, and A. Craig. "The IT Function and Robotic Process Automation", 2015. Available from:

[27] Hammer, M., "Reengineering Work: Don't Automate, Obliterate.". Harvard Business Review, 1990. 68 (4): p. 104-112.

[28] Myers, M.D. and M. Newman, "The Qualitative Interview in IS Research: Examining the Craft". Information and Organization, 2007. 17(1): p. 2-26.

[29] Hoff, K.A. and M. Bashir, "Trust in Automation: Integrating Empirical Evidence on Factors that Influence Trust". Human Factors, 2015. 57(3): p. 407-434.

[30] Eyers, J. "CBA Trials Blockchain for Trade Finance", 2016 [cited 07.07.2020]. Available from: https://bit.ly/2KkDS4j

[31] Sadiq, S.W., O. Marjanovic, and M.E. Orlowska, "Managing Change and Time in Dynamic Workflow Processes". International Journal of Cooperative Information Systems, 2000. 9(01n02): p. 93-116.

[32] Torres, L.A. "Change Healthcare and TIBCO to Bring Blockchain-Powered Smart Contracts to Healthcare", 2018 [cited 06.07.2020]. Available from: https://bit.ly/32N4Vf1

[33] Teepe, W., R. van de Riet, and M. Olivier, "Workflow Analyzed for Security and Privacy in Using Databases", in Data and Application Security., T. B., et al., Editors. 2000, Springer: Boston, MA. p. 271-282.

[34] Wang, E.T. and J.H. Chen, "The Influence of Governance Equilibrium on ERP Project Success". Decision Support Systems, 2006. 41(4): p. 708-727.

[35] Daley, S. "Nine Companies Using Blockchain to Revolutionize Insurance", 2019 [cited 06.07.2020]. Available from: https://builtin.com/blockchain/blockchain-insurance-companies

[36] Catlin, T. and J.-T. Lorenz. "Digital Disruption in Insurance: Cutting Through the Noise", McKinsey & Company, 2017. Available from: https://mck.co/2Uz2jwL.