Leiden University

MSc. ICT in Business

Blockchain Technology and Its Implementation within Enterprise Information Systems

Name:	Abdulhadi Abulzahab
Student ID:	1688421
Date:	01/July/2017
First Supervisor:	Dr. Hans Le Fever
Second Supervisor:	Tino de Rijk

MASTER'S THESIS

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Blockchain Technology and Its Implementation within Enterprise Information Systems

Abdulhadi Abulzahab
S1688421
a.abulzahab@umail.leidenuniv.nl

In partial fulfillment of the requirements for the degree of:

Master of Science (M.Sc.) of ICT in Business

Graduation: July 2017

Supervisor: Dr. Hans T. Le Fever

Second reader: Tino de Rijk

Leiden Institute of Advanced Computer Science (LIACS)
Leiden University
Niels Bohrweg 1
2333 CA Leiden
The Netherlands

Acknowledgement

First, I would also like to express my profound gratitude to the thesis supervisor Dr. Hans Le Fever for his valuable guidance and feedback through the process of writing this thesis. Despite his busy schedule, he always allowed room for meetings and discussions, which helped steering this research in the right direction whenever needed.

Special thanks to Mr. Tino de Rijk, as the second reader of this thesis. His thorough review was valuable, and helped in finalizing this research and enrich its arguments.

Finally, thanks to the experts who were involved in the design and validation process of this research project. Without their passionate participation and input, the validation process could not have been successfully conducted.

Author

Abdulhadi Abulzahab

Abstract

Blockchain is the core technology upon which bitcoin and other digital currencies were built. It is a distributed ledger, similar to a sequential database, to which all participants have access. Using consensus and encryption protocols, blockchain could create a trust less ecosystem to transfer value amongst parties without central authorities. The technology is promoted based on three main characteristics: 1) Decentralization, 2) security, and 3) immutability. Based on these advantages, many authors shed their light on promising use cases in different industries, but the focus was mostly on the financial market. Literature review also revealed that the revolutionary attribute of blockchain is overemphasized and the hype created around it is not supported by sufficient empirical studies, which makes it difficult for organizations to understand where and how blockchain can be implemented in their value chain. Findings from the preliminary interviews with blockchain experts, suggested that in order to create collective understanding of this technology, while aiming at accelerated adoption, enterprises should consider integrating a blockchain within their existing enterprise systems. Blockchain platforms will not stand alone, but they will function within the core of multiple, increasingly distributed ecosystems. However, blockchain creates a system that does not need a trusted party while the trusted party in the domain of the enterprise is already there, namely the enterprise itself. Furthermore, databases used by the traditional enterprise system are heavily protected. They also provide robustness and high performance. Therefore, the only remaining differentiator for blockchain is immutability, which might not always be a preferred attribute for business applications.

This research follows a design science methodology, which will lead to designing an artifact for a blockchain solution that fits within the domain of enterprises. Following this approach, we will analyze both the value system of blockchain, and the business environment, including its supporting IT infrastructure. Interviews with ERP experts helped in identifying real-world challenges, mainly in data integration and interoperability, which contributed to the construction of a detailed scenario where blockchain can be implemented to address those challenges. Moreover,

since blockchains will not stand alone, we will also provide an architecture of a customized software connector to integrate the blockchain platform with existing systems. Eventually, both designs were checked by experts to evaluate their validity and some refinements were made before presenting the final design. As a result, we propose blockchain as a synchronization mechanism between different databases in distributed enterprise systems. The artifact will increase efficiency and facilitate interoperability between different functions and business partners. In addition, the immutable records maintained by the system will play a significant role in financial audit processes and improve trust with legal authorities.

Table of Contents

List of figures

List of tables

List of abbreviations

API:	Application Programming Interface
Bitcoin	Bitcoin as A System
bitcoin/ BTC:	Bitcoin Currency.
CRM:	Customer Relationship Management
DTCC:	Depository Trust & Clearing Corporation.
DTL:	Distributed Ledger Technology.
EAI:	Enterprise Application Integration
EIS:	Enterprise Information System
ERP:	Enterprise Resource Planning.
ES:	Enterprise System
ESB:	Enterprise Service Bus
FICC:	Fixed Income Clearing Corporation.
GDP:	Gross Domestic Product
GL:	General Ledger.
GRN:	Goods Received Note
GUI:	Graphical User Interface.
HCM:	Human Capital Management
IC:	Inventory Control Systems.
KMS:	Knowledge Management System
KYC:	Know Your Customer.
MCF:	Managed Connection Factory
MDM:	Master Data Management
MOM:	Message Oriented Middleware
MRP:	Manufacturing Resources Planning
PBS:	Packet-Based Solution
PO:	Purchase Order
POW:	Proof of Work.
PR:	Purchase Request
RPC:	Remote Procedural Call
SCM:	Supply Chain Management
SI:	System Integration
SOA:	Service Oriented Architecture
UBL:	Universal Business Language
WOOM:	Web-Based Object-Oriented Model.

Chapter 1
Introduction to the research

This chapter introduces the research. In the following subsections, a general introduction is followed by explanation of the problem statement, which is supported by literature and forms the basis for the research objectives. After the research questions and the scope are identified, the research methodology together with the academic approach followed to answer these questions are introduced.

1.1 Introduction

Bitcoins is the most successful implementation of blockchain. Bitcoin could achieve its success in creating a trust mechanism and remove the need for a trusted third party to exchange value and complete transactions. Simply put, Blockchain is a public distributed ledger, similar to a database, to which everyone has access and has the ability to add and verify records. It holds immutable records of all transactions that have ever happened in the network. Thus, it allows every user to verify the fact that every specific transaction has indeed taken place at a specific moment in time. The discovery of blockchain importance as separated technology from Bitcoin can be dated to 2015 when the majority of financial institutions publicly announced interest into the innovation (Glaser, 2017).

Blockchain is one of the most popular topics on the web. Google trends show that searching for Blockchain has risen 1900% since 2013 until the beginning of 2017. Authors from MIT identified blockchain technology to be the fifth horizon of networked technology. It has the potential to create a wave of innovation across multiple industries and disrupt the current business models (Shrier, Sharma, & Pentland, 2016). Venture investment in the field has grown to $1.11 billion in 2016 (WeUseCoins, 2017). Public and private sectors have begun to experiment how blockchain can be applied to tackle some issues in transparency, security, and trust. Although some progress has been made, there is yet no commercialized blockchain solutions, rather than digital currency.

To this day, Bitcoin is still the most common application using Blockchain technology ("Crypto-Currency Market Capitalizations,"). The number of transfers and users in the Bitcoin network is constantly increasing (Kondor, Posfai, Csabai, & Vattay, 2014). Blockchain has proven viable for digital currency exchange. In addition to Bitcoin, many other crypto currencies "Altcoins" have emerged. Providing different protocols and currencies but using the same underlying principles of Blockchain, altcoins until today count for more than 650 different currencies ("Crypto-Currency Market Capitalizations," n.d.).

Interestingly enough, intermediaries that are at risk to be eliminated by the technology, have taken the lead in investigating it. Especially after the introduction of smart contracts in 2014, which facilitate more capabilities than simple transfer of value, major financial institutions and governments began to realize that the revolution resides not in Bitcoin but in the underlying technology, which has become one of the most discussed and allegedly disruptive innovations (Glaser, 2017). Although it is still in its early adoption phase, many new ventures around the world have already been funded to experiment the technology stack at all levels. In 2016, many international consortia have been formed to create proof-of-concepts (POCs), which are blockchain platforms customized to their protocols, privacy and consensus mechanisms according to industry standards. The generic concept called Distributed Ledger Technology (DLT) and Blockchain is an instance of it.

To this end, there are around 20 consortia around the world. Over 70 international banks have joined in one consortium called R3 CEV, which is the largest in the world. R3 was originally designed to investigate and roll out distributed ledger technology to global banking, leveraging Blockchain technology in a private, centralized environment to harmonize and add efficiency to a range of internal and external processes (Gendal Brown, Carlyle, Grigg, & Hearn, 2016). IBM and Microsoft are already providing distributed ledger in their cloud services and Linux Foundation has launched an open source blockchain project called "Hyper Ledger" built upon the Ethereum platform (Linux Foundation, 2015). Each blockchain solution provides different consensus mechanisms, encryption algorithms, and/or auditing permissions.

The initial impact, anticipated by (Andrews & Manuel, 2016), will likely be related to financial industry. However, researchers and practitioners from other industries are investigating the technology and its possible use cases. Yet, the design of a system that implement blockchain technology has not been systematically explored and there is little understanding about the advantages provided by such a system over the currently implemented ones. Providing such a level of understanding is necessary for organizations to meet the challenges posed to blockchain adoption and implementation. This thesis moves forward in this direction.

It is worth mentioning that in this paper we will mention "Blockchain" and "DL" interchangeably. We are actually referring to the same concept. Blockchain is a type of Distributed Ledger that is used by Bitcoin. However, in case we are referring to Bitcoin's implementation, we will explicitly mention it.

1.2 Structure of the paper

This report is structured along five chapters, following a Design Science approach (Hevner & Chatterjee, 2010). The first chapter introduces the problem statement, the scope, and the methodology followed in the research. The framework of the research requires handling three aspects; knowledge base, business environment, and system design. These are the second, the third, and the fourth chapters respectively. The second chapter provides analysis of the knowledge base, where we analyze the current knowledge about the topic. This analysis is divided into two subsections; theoretical background, which are facts about blockchain technology and it components, the second subsection reviews the literature about the topic and different opinions and views are presented. The third chapter analyses the business environment. This include business problems and opportunities in addition to comprehensive technological analysis of the enterprise systems, their functional and technical infrastructure in addition to the advantages and challenges. The fourth chapter in this paper comprises our results and findings. We present in this chapter our proposed artifact and provide evaluation arguments based on knowledge base and expert's interviews. Finally, the fifth chapter presents discussion about the problem and our findings, conclusions, and recommendation for future research.

1.3 Problem Statement

The strong interest in Blockchain technology can be observed by the rapid increase in publications, conferences, and journal articles in the last two years. Despite its popularity, the number of empirical studies that have been published on the topic is remarkably low and search results in academic libraries are minimal. Most of the papers talk about the revolutionary nature of the technology and that it has the potential to disrupt multiple industries and business models, which created a hype around its utilities and use cases. Furthermore, Blockchain enthusiasts have and still are formulating significant number of use cases. The available literature can be categorized as first, technical papers that explain the technology stack and propose alternative protocols to improve scalability and increase speed by providing alternative cryptography algorithms or trust mechanisms. These white papers are not concerned with the environment where the system is to be implemented and do not propose use cases. The second category of literature contemplate use cases at an abstract level without investigating the technological validity. The focus is mostly on financial industry and its related services. The likely reason for this is the potential disruption of Blockchain anticipated by major financial institutions, consulting companies and even government (Walport, 2016).

Nevertheless, all these publications create greater ambiguity and cause more confusion to the reader. The 'how' and 'why' are often missing. This was also emphasized by (Glaser, 2017) where he explained how the ecosystem introduced by blockchain leads to confusion about the actual use cases and their technological and economical validity. Moreover, a paper published jointly by two universities in Finland and Sweden, (Lindman, Tuunainen, & Rossi, 2017) urged empirical research by academia to enrich the knowledge base about what is described as "the most revolutionary invention since the advent of the Internet". The European Central

Bank concluded that the distributed ledger is a very new technology and it is therefore still difficult to predict exactly what kind of an impact it will have on the market infrastructure and that deserves further investigation (Pinna & Ruttenberg, 2016). The same conclusions found in most publications. Furthermore, interesting question was whether blockchain solution would provide better, more secure, solutions than what is already existing (Walport, 2016).

(Morabito, 2017) concluded that the technology is nascent and the realization of its potentials to overcome the existing issues in transactions among businesses as well as to improve current business practices, need to be in depth investigated. He also stated that researchers must properly address other concerns such as the regulations that govern how the system works, security and privacy issues, integration concerns and cultural acceptance. In addition, proper research, management, and experience are required to understand fully the business domain as well as how blockchain technology can fit and meet business requirements.

1.4 Research Objectives and Questions

Enterprise Systems (ES) are described by (Duan, Faker, Fesak, & Stuart, 2013) as key element of infrastructure that maps all processes and data of an enterprise in order to deliver business solutions. Blockchain, in comparison, is an infrastructure that maps all kind of data and uses smart contracts to automate business processes and enforce business logic. Furthermore, the introduction of the cloud and cloud computing in the last decade, had remarkable impact on IT infrastructure of organizations around the world (Peng & Gala, 2014). Most organizations migrated part of their IT solutions to the cloud. According to 2017 survey run by (Weins, 2017) 85 percent of surveyed organization have plans to use multi-cloud solutions for their IT strategy. As a comparison, the Distributed Ledger Technology is considered the next wave of networked technology after the cloud (Shrier, Sharma, et al., 2016). The disruptive characteristic of blockchain technology seems to be over emphasized, whilst potential business value and advantages over currently available solutions have not been fully explored and discussed in either the industry or academia (Lindman et al., 2017). The hype created around the technology and the lack of empirical studies to support the envisioned use cases, made it difficult for organizations to understand how they can implement blockchain in their value chain. Authors argue that in order to get the full benefits of blockchain, it should be implemented in a consortium (Deloitte, 2016; Young, 2017), but it has been also stated that blockchain will deliver benefits if it is implemented within organizations (Drane, 2016).

In fact, all enterprises use systems that support business processes, information flows, reporting, and data analytics. While ES are generally centralized packaged-based solutions (PBS), they can also be distributed, custom developed systems created to support a specific organization's needs (He & Da Xu, 2014).

The purpose of this research is to investigate the advantages of DLTs over the current ERP systems and whether they can replace /or be integrated with them. Its aim, in particular is to seek in-depth insights from highly experienced IT professionals to identify and investigate potential benefits or possible areas for integrating the distributed ledger with enterprise systems, as well as to provide useful lessons to help companies better prepare themselves for potential IT transformation. Moreover, it will help us to understand the implementation, implementers, users, and their respective objectives. This will lead us to the main questions of this thesis:

1- **Can blockchain, or the distributed ledger technology, fit within a centralized, trusted ecosystem like Enterprise systems and if so, where?**
2- **What are the advantages provided by blockchain technology over the current enterprise systems?**
3- **Which type of business processes will be affected by this integration?**

1.5 Research Methodology

The general approach followed in this thesis is Design Science approach for Information System. We refer in our work to the guidelines introduced by (Hevner & Chatterjee, 2010; Von Alan, March, Park, & Ram, 2004). They argued that the critical nature of design science research in Information System lies in the identification of new information technology capabilities, resulting in the expansion of

IS into new realms. Such a result is significant IS research only if there is a serious question about the ability to construct such an artifact; this is uncertainty about its ability to perform appropriately. In the same matter, many authors raised questions about where Blockchain can be applied and whether it is a better solution than the already existing solutions. Therefore, the product of this research would be an IT artifact that addresses the uncertainty problem by organizations about the feasibility of applying blockchain in their value chain. The framework upon which we will base the research is depicted in (Figure 1-1).

The environment in our case defines the problem space of the phenomena of interest where the problems, opportunities, organization context and business processes that define business needs (Simon, 1996). For this research, we will examine the existing infrastructure, information systems, applications and communications architecture that might be replaced by or integrated with any possible new artifact (Hevner & Chatterjee, 2010). Due to the broad variety and complexity of enterprise systems, we will focus on the general architecture and the generic components that must exist in any system. The same is also applied to business processes; the focus will be on a generic business problem in an organizational context and the opportunity presented by the proposed artifact.

In the environment analysis, the paper will study the business side and the technology side as well. The analysis includes IT infrastructure that supports the business processes in scope, their applications, and communication architecture.

Knowledge base analysis helped in acquiring the materials necessary to construct the product. We will explore literature about the technology stack of blockchain, smart contracts, and cryptography, in addition to reviewing the literature about Enterprise Systems and application integration. As stated in (Easterby-Smith et al. 2008 cited by (Saunders, 2011)), about deductive approach for new phenomena "researchers in this tradition are more likely to work with qualitative data and to use a variety of methods to collect these data in order to establish different views of phenomena."

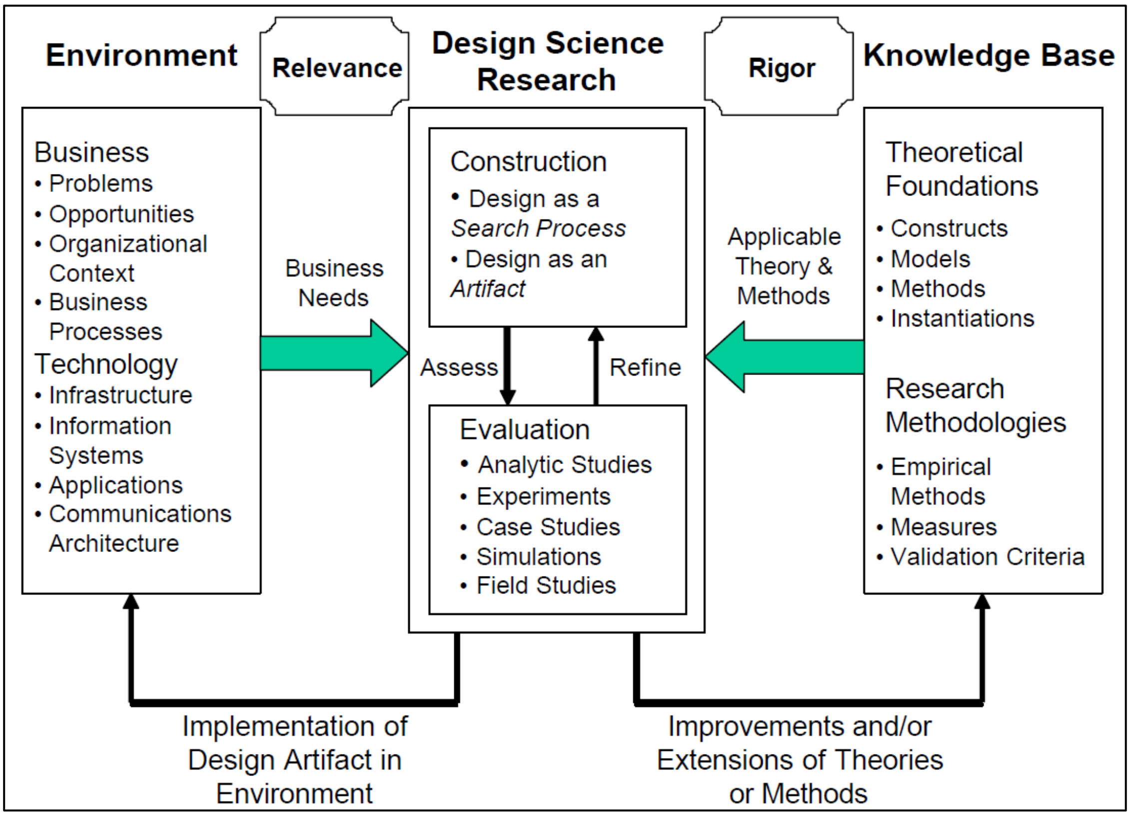

Figure 1-1 Design Science Research Framework – source: (Hevner & Chatterjee, 2010)

Therefore, the author collected the data from multiple sources including academic libraries, white papers, and other publications. In order to stay up to date, the author subscribed to newsfeed and blogs about the topic, attended webinars, and many conferences and meetings organized across The Netherlands. This helped in yielding an understanding of the phenomena, latest advances, challenges, application domain (e.g. requirements and constraints), and the solution domain (e.g. technical and organizational).

The literature review process began at early stages of the research from broad perspective on blockchain technology and its current implementations. Reviewing academic papers helped in identifying the current state of the research about the technology and refine research ideas. By following citation, we could go deeper into the literature, and define keywords for searching, which helped us to develop sharper questions and more insights in the area of investigation. This made it possible to identify the gap in the knowledge base, which we are trying to fill through this thesis.

Although reviewing the literature was an early activity of the research but it continues through the lifecycle of the project. For reliability purpose, it was necessary to review literature form multi-disciplines like IT, business, finance, and governance.

Construction and evaluation of the artifact are parts of iterative process in order to reach a solution that satisfies the laws existing in the environment (Simon, 1996). Thus, creating the artifact started with initial interviews with experts to get more understanding of business processes and real-world problems faced by consultants. In order to develop an effective solution, we made a preliminary design based on the knowledge we gained from studying the technology. Afterward, we conducted interviews to introduce the design, validate it, and refine it. Expertise of interviewees was useful for the assessment of technical and functional validity of the artifact. Thanks to their contribution, we could reach to what we believe is a plausible end (see Chapter 4). (Table 1:1) lists the interviewees through the project. There were two types of interviews 1) preliminary, informal, unstructured interviews and 2) validation, formal, semi-structured interviews. The purpose of the preliminary interviews was to discuss the topic of the research and identify its feasibility. Whereas, validation interviews were intended to validate and refine the designed architecture.

Type of interview	Interviewee	Position
Preliminary	J. W.	Blockchain consultant
Preliminary	E. S.	Blockchain researcher
Preliminary	W.D.R.	Procurement consultant
Preliminary	L. V. D.	ERP functional consultant
Validation	P.J. D.V.	Application integration consultant
Validation	E. H	Financial Auditor, corporate control
Validation	R. P.	Blockchain developer
Validation	S. B.	ERP functional consultant

Table 1:1 List of Interviewees

We contribute to the knowledge base by the creation of an architecture for implementing blockchain technology within enterprise systems, demonstrating the capabilities and limitation of such an artifact, evaluate, and predict its potential benefits along with the impact on business processes and organizational changes, in addition to the intermediaries' roles in the process. We will demonstrate this by describing one existing business process using the artifact as an example that can be generalized to other processes.

Chapter 2
Knowledge Base Analysis

Knowledge base analysis is the study of available literature about the topic. This analysis is divided into two sections; theoretical background, which are facts about blockchain technology and it components, the second section reviews the literature about the topic in terms of latest developments and authors' opinions and views.

2.1 Theoretical background

2.1.1 Bitcoin:

On October 2008, a person named Satoshi Nakamoto posted in a mailing list for cryptographers an introduction to his thesis titled "*Bitcoin: A Peer-To-Peer Electronic Cash System*" about new electronic cash system. He stated that the new system has the following characteristics (Nakamoto, 2008b):

- A peer-to-peer network that prevents double spending.
- No mint or other trusted parties.
- Participants can be anonymous.
- New coins are made from Hashcash style proof-of-work (POW).
- The proof-of-work for new coins generation also powers the network to prevent double spending.

Bitcoin is digital cash, a digital currency and online payment system comprise a public transaction ledger (Nakamoto, 2008a). Bitcoin cleverly combines existing contributions from decades of research. It also solved fundamental problems in a highly sophisticated, original and practically viable way; it uses a proof of work scheme to limit the number of votes per entity, and thus renders decentralization practical (Tschorsch & Scheuermann, 2015). Within bitcoin, encryption techniques, are used to regulate the generation of units of currency and verify the transfer of funds, operating independently of a central bank (Swan, 2015).

The role of trusted third party is to validate, safeguard, and preserve transactions. A certain percentage of fraud is unavoidable in online transactions and that needs mediation by financial transactions. This results in high transaction costs (Crosby, Pattanayak, Verma, & Kalyanaraman, 2016).

To eliminate the need for a bank, the ledger holding ownership records, must also be distributed. Since the records are digital and distributed, it is possible that someone runs two transactions at the same time for the same digital coin to different recipients, which is called "double spending." In traditional scenarios, this cannot happen since the bank is able to detect and prevent the actions from proceeding. Distribution of information and the problem of mutual agreement on a consistent state is a challenge, especially in the presence of selfish and/or malicious participants (Tschorsch & Scheuermann, 2015). This pushed the idea to create quorum systems, which accept the possibility of faulty information and the existence of malicious entities in such a cooperative environment. Therefore, a voting system was conceptualized, where the majority of peers in the system must agree on the correctness of the entrees. This is provided by the underlying ledger, the Blockchain.

In the early 2009, the first transaction ever in Bitcoin was recorded and the first BTC (Bitcoin currency) was issued (Nomura Research Institute, 2016). Since 2011 until today, the number of bitcoin transaction is doubling and market capitalization has reached just above sixteen billion US dollar (Blockchain, n.d.).

2.1.2 Bitcoin's blockchain

Blockchain is the core technology upon which Bitcoin was built, and later become known as Distributed Ledger Technology (DLT). It was first used by Bitcoin therefore it is closely related to it and because of it, bitcoin could achieve its success in creating a trust mechanism and remove the need for a trusted third party. It is a public distributed ledger, similar to a database, to which everyone has access and has the ability to add and verify records. It holds immutable records of all transactions that have happened in the network. Thus, it allows every user to verify the fact that every specific transaction has indeed taken place at a specific moment in time.

Anyone can download the open source software of Bitcoin and become a node in the network. Transactions are done between nodes using public and private key cryptography and IP addresses in order to verify the sender and the receiver. The sender and the receiver need to trust each other and complete the transaction successfully without intermediary. Therefore, transactions in Bitcoin are publicly announced and privacy is preserved based on the concept proposed by (Dai, 1998). The trust in the network is achieved by consensus rather than by individual nodes. In this way, every node can represent the bank, or central authority, by holding a copy of the ledger, which would classically be stored at the central bank.

When the owner of a coin wants to send it to another node, he uses his private key to encrypt the coin and sign the transaction. The receiver is identifiable by their IP address, and identifies the sender by their public key that corresponds to the private key used to sign the transaction. Once the receiver is notified about the transaction, and before he accepts it, he broadcasts it to the network. To ensure the legitimacy of the transaction i.e. the sender has not spent the same coin(s) before "double spending" and has enough coins to issue the transaction, Bitcoin addresses this problem by letting the entire network verify it before recording it in the public ledger. However, these assurances come with a price of latency- a prototypical distributed consensus problem (Tschorsch & Scheuermann, 2015). Every approximately ten minutes, a group of transactions are verified and put in a block and then the block is linked to the previous one to form a chain (Figure 2-1). This is why it is called "*Blockchain*." Participants on the network verify the transactions that have been broadcasted. Blockchains are immutable, which means that once transactions have been approved and registered, they cannot be modified. This is one of the reliability enforcement features in Blockchain.

There can be multiple blocks created by different nodes at the same time. One cannot rely on the order since blocks can arrive at different orders at different point in the network (Crosby et al., 2016). Bitcoin solves this problem by introducing a mathematical puzzle: each block will be accepted in the chain if it contains a solution for a special mathematical problem introduced by the system. This is also known as "*proof of work*."

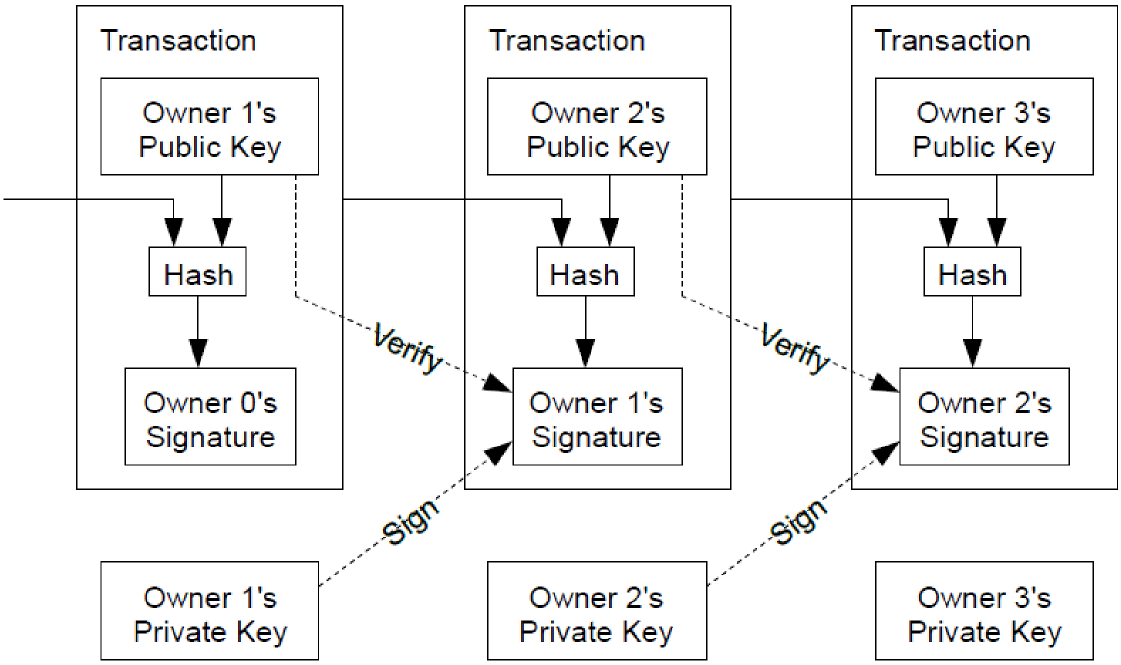

Figure 2-1 Blockchain Concept – source: (Nakamoto, 2008a)

2.1.3 Proof Of work (POW)

POW is the underlying mechanism of providing consensus over transactions, or value transfer, between nodes in a Bitcoin network. In POW, a node generating a block needs to prove that it has put enough computing resources to solve a mathematical puzzle. For instance, a node can be required to find a nonce which when hashed with both transactions and hashed of previous blocks produces a hash with a certain number of leading zeros. This is called "mining." The average effort required is exponential in the number of zero bits required but the verification process is very simple and can be done by executing a single hash(Nakamoto, 2008a).

Nodes on the network devote their computation resources for mining. Bitcoin provides an incentive mechanism to stimulate more nodes to participate in mining process; the first miner to find the solution will be rewarded with a few bitcoins. To prevent double spending, every ten minutes mining computers collect a few hundred pending Bitcoin transactions, which will comprise a new block (Figure 2-2). Thus, the same person cannot spend the same coin twice at the same time. Before a transaction is registered in a block, its status remain "unconfirmed."

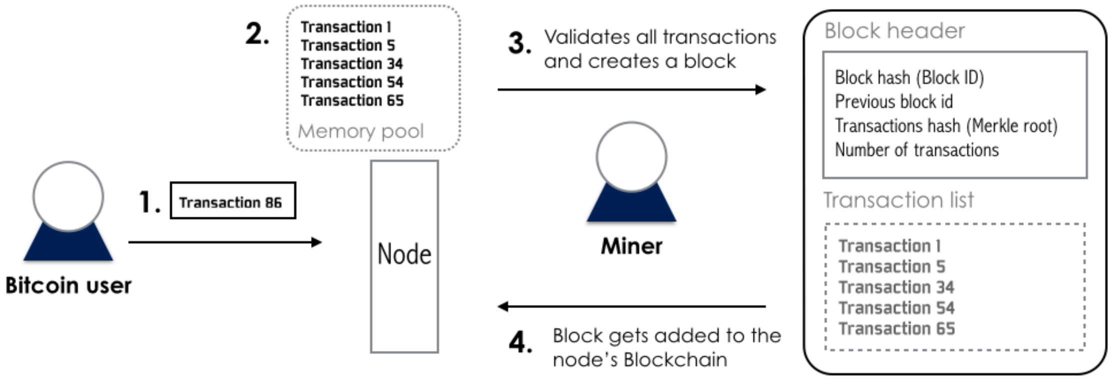

Figure 2-2 Proof of Work – source: tech.eu

The first miner to find a solution for the hash, announces his solution to other miners, which then check the validity of transaction, i.e., weather the sender has the right to spend the money and weather the hash is a correct solution. If enough of them grant their approval, 51% or more, the block is cryptographically added to the ledger and miners move on to the next set of transactions (Shrier, Sharma, et al., 2016). The consensus idea was first introduced in the concept of Hashcash by (Back, 2002), but the consensus would be formed by entities on the system instead of computational power, which led to the Sybil attack (Douceur, 2002). This mechanism solve the problem of double spending and makes it difficult for bad players since they need to have more than the half of the mining capacity to be in control.

Linking blocks together in a chain through their hashes, solves the well-known Byzantine Generals Problem, which describes the reliability of computer system when coping with the failure of one or more of its components. A failed component may exhibit a type of behavior that is often overlooked by sending conflicting information to different parts of the system. This was explained abstractly as the Byzantine Generals Problem by (Lamport, Shostak, & Pease, 1982).

The 10 minutes block creation time was chosen as a tradeoff between confirmation time and the amount of work wasted due to chain forks[1]. But it also ensures to reach every corner of the network even in the face of prolonged propagation times (Tschorsch & Scheuermann, 2015). Therefore, it synchronized the network loosely after a while.

This suggests that Bitcoin underlies the assumptions of a synchronous network and is more tolerant than the deterministic constraint and takes eventual consistency as adequate. As long as more than the half of the hash power is controlled by honest miners, the network will eventually reach consensus, even in the presence of malicious miners (Tschorsch & Scheuermann, 2015).

Some of the technical limitations of Bitcoin are (Nomura Research Institute, 2016; Swan, 2015):

- The amount of transactions processed per unit time is small; seven transactions per second compared to two thousands transactions per second in e.g. the VISA credit card network.
- The size of one block is limited to 1 MB and a block is created every 10 minutes.
- Although participants are anonymous, they can be identified by their public key and it would not be difficult to track the IP address. In addition, transaction details are disclosed and privacy may not be protected.
- Mining requires powerful computation power, which makes mining exclusive to some nodes.
- Excessive power is consumed due to the proof of work.
- Timestamps affixed to transactions within a block are neither accurate nor guaranteed.
- Transaction fees are dependent on the mining time rather than percentage or fixed fee.

[1] **Chain forks** happen when two blocks are created at the exact same time in the network when two miners find a solution. Therefore, there will be two blocks connected to one block. This is rare situation, but when it happens, the chain will continue after only one of them. The block which is left on the side, is called "orphan block" and transactions in it will remain unconfirmed. Thus, extra work is needed to rewrite the transactions again.

2.1.4 Proof of Stake

Proof of stake is an alternative protocol for reaching consensus in the network. The mining rights in this mechanism is granted to participants in proportion to their holding of the currency with the blockchain network. Miners need to prove the ownership of a certain amount of currency to mine blocks. The duration of holding coins, coin's age, plays also a role in the process. Therefore, protection of the chain from malicious attack is guaranteed. Attackers need to own large amount of currency, which is expensive. Besides, someone with large stake would be careful about the chain safety, which will have an impact on the value of the currency.

Miners in Proof of Stake do not need computation power for mining or solving the puzzle. This make mining rights equally distributed on network participants and the reward is proportionally divided. In addition, block creation does not take ten minutes because consensus is reached faster than Proof of Work (Tschorsch & Scheuermann, 2015).

Other protocols are used by other crypto-currencies like proof of burn or proof of activity. They all use different consensus mechanisms but still use the same underlying blockchain principle.

2.1.5 Smart contracts

The phrase "smart contracts" was first introduced by Nick Szabo in 1994. This term emphasized the goal of applying contract law and related business practices to the design of e-commerce protocols between parties over the Internet. According to (Szabo, 1997):

"A smart contract is a computerized transaction protocol that executes the terms of a contract. The general objectives are to satisfy common contractual conditions (such as payment terms, liens, confidentiality, and even enforcement), minimize exceptions both malicious and accidental, and minimize the need for trusted intermediaries. Related economic goals include lowering fraud loss, arbitrations and enforcement costs, and other transaction costs."

The word transaction, if we will consider blockchain as a database, means the update of records in a table. In the case of crypto-currencies, the update is about transferring the coin from one owner to another. However, developments in blockchain algorithms allowed the coins to represent other values than money. Where transaction then refers to the transfer of tokens from one user to another. In 2013, a new implementation was proposed by Vitalik Buterin, which would enable building in a scripting language, similar to stored procedures in traditional databases, to execute additional business logic triggered by a transaction (Glaser, 2017). The idea was to build on existing concepts, such as Bitcoin, and improve upon transactional speed and overall security. The first blockchain platform that provides this feature is Ethereum. It extends the basic idea of a scripting language by integrating a fully-fledged programming language executed by an internal "Ethereum virtual machine" (Wood, 2014). These programming languages have also access to complex data types, data structures, and even small, locally separated databases that can be used to store and retrieve data (Glaser, 2017). The scripting code is built in the blockchain system and copied to every node in the network. According to the conditions upon which nodes of the system have agreed, the code can be executed automatically when a specific event occurs. This new implementation would allow more possibilities and capabilities of Blockchain than crypto-currencies were intended to do. These pieces of code are referred to as smart contracts. The introduction of smart contracts is considered as starting point of a new version, Blockchain 2.0.

Blockchain 1.0 is a platform used for the decentralization of money and payments, programs are built upon to perform simple transfer and exchange operations. Whereas Blockchain 2.0 is a platform, upon which programs that are more sophisticated can be deployed. These programs inherit the same characteristics of blockchain; they are permanent, they cannot be altered, they have control over assets in the network, and they will always execute as written and one cannot interfere with their operations.

The term "contract" does not only comes in the legal context. In blockchain environment, it also means "autonomous operations," a code that articulate, verify and enforce an agreement between parties (e.g. transfer value automatically on a specific date); and “governance application” (e.g. if this … then ….). Calling these programs contracts is helpful in that this code is governing something important or valuable and in many cases, smart contract code in not used in isolation but as a small piece in a larger application (CoinDesk, 2016). They can themselves hold balances or even control other smart contract programs. (Figure 2-3) depicts how smart contract are implemented on blockchain.

In legal and financial contexts, smart contract refers to a way of using blockchain to complement or replace existing legal contracts. These contracts use code in combination with traditional legal language. This has been attractive to financial institutions around the world to develop and experiment in blockchain as it holds the potential to formulate smart contracts for financial instruments like bonds, derivatives, and shares. Achieving more automation and simplifying many processing intensive systems related to trading and servicing of these processes (CoinDesk, 2016). (Table 2:1) shows the current offerings of smart contracts.

	Blockchain without smart contracts	**Blockchains with smart contracts**	**Blockchain with native smart contracts**
What?	Dispersed storage	Dispersed compute: holds the capacity to compute predefined logic	Dispersed compute: holds the capacity to compute any logic
Examples	Bitcoin (public) Litecoin (public) Multichain (private)	NXT (public)	Ethereum (public) Eris(private) Clearmatics (private)

Table 2:1 Various smart contracts offerings – source: (Lewis, 2016)

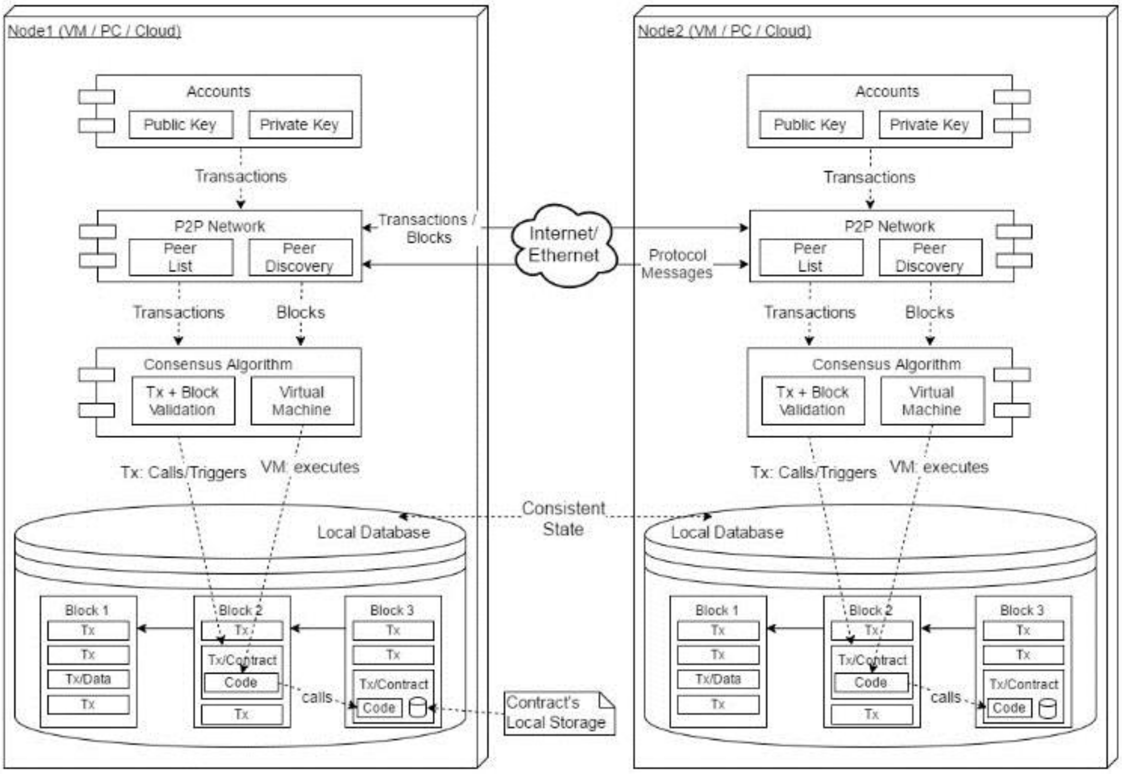

Figure 2-3 Smart Contracts – source: (Wood, 2014)

2.1.6 Public, consortium and private Blockchains

The development of smart contracts and cryptographic consensus algorithms facilitated the emergence of customized blockchain platforms by exploiting some principles and discarding or altering others.

Public blockchain is Bitcoin's blockchain-like. Anyone with an internet connection can download the software and become and node in the network. By joining the network, the participant is able to read and send transactions and expect to see them included, if they are valid. They are also able to mine and validate transactions. Participants in a public blockchain do not have to provide any personal information or identification. They are only identified by their public key. Therefore, public blockchains are anonymous although there are concerns about the possibility of deriving the associated IP-address.

In addition, users on the network are protected even from the developer of the platform since any changes in the technical properties of the system need to be approved by the majority of the nodes. Thus, no one is able to control balances, revert confirmed transactions, or even give a way coins out of nowhere. Therefore, public blockchains are decentralized.

Consortium blockchains are custom types distributed ledgers. Joining the network requires that participants are member of the consortium. These network vary between permissioned and permission-less.

Permission-less consortium, is like public blockchain but only joining the network is restricted. Whereas, permissioned blockchain indicates that there are more restrictions on nodes privileges. For instance, validate and read permissions can be attached to each transaction where a smart contract can control who can validate or read transaction. It is also possible that validating transactions is granted only to pre-appointed nodes. There also may be some parts of the blockchain publicly available. These blockchains are generally considered partially decentralized.

Advantages provided by this type of blockchains are:

- Participants are known and limited, which makes it easy to reach consensus without the need for proof of work or other expensive computation power.
- Nodes are trusted to be very connected, and faults can quickly be fixed by manual intervention.
- Due to the ability of controlling read permissions, private blockchains can provide a greater level of privacy.
- There is no need to use coins. The main idea of consortium blockchain is to be able to transfer different kind asset classes. Because nodes are known and registered it is possible to connect physical assets to transactions on the chain.
- Similarly, transactions need not to be grouped in blocks. Instead, multi-signature is used to validate transactions and record them on the chain. This overcomes the limitation of the block's size and the amount of data that can be stored.

Private Blockchains are similar to permissioned consortium blockchains, but these platforms are owned and implemented by one single organization. This type of blockchain is fully centralized. Validation is done by pre-determined nodes.

Owning organization can optionally grant permissions to any external permissions.

2.2 Literature review

Since its introduction in 2008, Bitcoin was considered revolutionary invention in the payment sector. The financial crisis of 2008 was one of the motives for Nakamoto to develop such a solution to free up the monetary from the control of banks (Shaw, 2016). Despite of the rapid increase of Bitcoin adoption financial institutions did not consider it as a threat although it was considered to be a disruptive innovation by many authors (Custers, 2015; Herrera-Joancomartí, 2015; Shrier, Sharma, et al., 2016; Trautman, 2016). Bitcoin will not replace fiat currency (Grau Miró, 2016; Huls, 2015; Millar & Brunet, 2016) due to price volatility. Legal entities will not adopt the technology because of anonymity of participants and other legal issues accompanied with decentralization.

(Kokalitcheva, 2015) stated that experts in the finance sector are weary of bitcoin price volatility, but see the blockchain as a promising technology. In an interview with McKinsey, Don Tapscott described blockchain as *the biggest innovation in computer science*. He said:

> *"This is an extraordinary thing. An immutable, unhackable distributed database of digital assets. This is a platform for truth and it is a platform for trust. The idea of a distributed database where trust is established through mass collaboration and clever code rather than through a powerful institution that does the authentication and the settlement"* (Rik Kirkland, 2016).

Blockchain technology started to attract the attention of experts from different industries in 2015, right after the introduction of smart autonomous contracts, which run natively on blockchain platform, introduced first by Ethereum. Blockchain was considered to be the fifth wave of disruptive computing paradigm (Shrier, Sharma, et al., 2016; Swan, 2015). Compared to the disruption caused by computer, the internet, mobile and cloud. Researches published about blockchain has started with modest number in 2013 but it surged in 2015.

The initial impact, anticipated by (Andrews & Manuel, 2016), will likely be related to asset classes where there is no central trading mechanism, such as Fixed Income Clearing Corporation (FICC) derivatives, syndicated loans and private investments.

Additional positive impacts identified by the World Economic Forum, is an increased financial inclusion in emerging markets, explosion in tradable assets, all kinds of value exchange can be "in theory" hosted on the blockchain, and better property records in emerging markets, and the ability to make everything a tradable asset.

Based on the attributes of blockchain, or the distributed ledger technology, authors (Ametrano, 2016; Beck & Müller-Bloch, 2017; Buterin, 2014; Pilkington, 2016; Swan, 2015; Wyman, 2016), envisaged applications and use cases for blockchain summarized in (Table 2:2).

Most of these use cases were also discussed by the UK Government Office For Science in their publication (Walport, 2016). The report contained recommendations and concerns about the maturity of the technology and the real benefits that can be reap from its on scale implementation. Important recommendations were that extensive research is required to ensure that distributed ledgers are scalable, secure and provide proof of correctness of their content. They need to provide high performance, low latency operations if they are to be implemented.

Notably, some of those papers avoided details about how such a system should work, or proposed using platforms that use digital currency, which is volatile and unreliable asset as mentioned earlier.

Purpose	Example
General	Value transaction, bonded contracts, third-party arbitration, multiparty transactions.
Financial transactions	Stock, private equity, crowd-funding, bonds, mutual funds, derivatives, annuities.
Public records	Land and property titles, vehicle registrations, business licenses, marriage certificates.

Identification	Driver's licenses, identity cards, passports, vote systems
Private records	Loans, contracts, wills, and trusts.
Attestation	Proof of insurance, proof of ownership, notarized documents.
Physical asset keys	Home, hotel rooms, rental cars, automobile access.
Intangible assets	Patents, trademarks, copyrights, reservations, domain names.

Table 2:2 Various use cases of DLT

Simone Tylor, vice president at Blockchain R&D at Barclays Bank, argues that it is highly challenging to deliver multiple complex services to one user. This has resulted in ever more complex data protection and data legislation measures to manage confidentiality and privacy of the individual. In addition, contracts and agreements have remained in paper form, rather than being automated in the wider economy. Combining the key attributes of a shared ledger (reconciliation through cryptography, replicated to many institutions, granular access control, and granular transparency and privacy). Smart contracts may create opportunities to address some of these challenges by allowing data to either be replicated or shared under specific conditions (Walport, 2016). In the report of (World Economic Forum, 2015), they expected that a tenth of global gross domestic product (GDP) will be stored on blockchain technology by 2027.

Smart contracts are being considered for a wide variety of uses, particularly for regulatory compliance, product traceability, service management, and also to defeat counterfeit products and fraud in various sectors (Nomura Research Institute, 2016; Swan, 2015; Walport, 2016). Initial application of blockchain technology is the original public ledger of Bitcoin, which has later inspired other implementations called alt-chains. These kinds of networks also provide trust-based services that are not limited to currency transactions. Bitnation.co, decentralized Non-Geographically Contingent Governance Service Aggregators (Bollen, 2013), offering a full range of legal services traditionally done by governments. This is also confirmed by Chris De Rose, he stated that the biggest boon to financial engineering comes in the form of smart contracts that are enforced electronically. The director of the Counterparty Foundation believes that smart contracts reduce risk and they cheaply ensure that all parts of a contract are fulfilled without interference (DeRose, 2016).

In addition to security, having only one irreversible version of truth is deemed valuable by many authors and practitioners in many industries. This was emphasized by (Shrier, Larossi, Sharma, & Pentland, 2016) as it is extremely beneficial to a settlement and reconciliation processes by reducing the number of parties, both external and internal, required to execute one transaction and working off a shared ledger.

FinTech Network with contribution from Rabobank and The Bank of New York Mellon, published a white paper containing four use cases for blockchain in different sectors (Taylor, Huls, & Mager, 2017):

- Fraud and Cyber-attack Protection: using DLT, not all information is located in one centralized database any more. It provides a historical record of all documents shared and compliance activities undertaken for each banking customer. Therefore, malicious attempts to view or change the data become part of the data itself, making hacks immediately obvious.
- Improve Know Your Customer (KYC) process: it is the process of identifying and verifying the identity of clients. Standard statements and compliance forms can be stored on blockchain so the same process would not be necessary by each financial institution for a client who is already verified by another accredited bank.
- Trading platform: blockchain can form a medium to exchange assets without intermediary. Blockchain would also address operational risk and administrative costs. Furthermore, the traceability and the permanent historic record that would exist on blockchain would provide assurance and authenticity all the way through the supply chain. In fact, the digital token is acting as a virtual "certificate of authenticity" which would have the advantage that it is far harder to steal or forge than a piece of paper.

- Payments: blockchain can be used to make payments in real-time globally, with real-time execution, complete transparency, real-time fraud analysis, and prevention at a reasonable cost.

All these publications and a lot more created a hype around the technology and its real use cases. Nevertheless, most of the publications do not provide details about how the system should work or how it should be implemented or adopted. Many controversial questions have not been clearly answered, in particular, questions about privacy, governance, business processes and the impact on the current systems in addition to arguments about the readiness of financial institutions to share their data.

Experimentations and literature continued extensively in 2016 to validate and propose use cases. It has been emphasized by (Buehler et al., 2015; Deloitte, 2016; PYMNTS.com, 2016; Young, 2017) that collaboration to develop a blockchain solution is the only way to realize the full benefits of the implementation and commercialization of Blockchain technology. Eric Piscini, a principal with Deloitte, stated:

"*Industry consortia will be critical to unlocking mass-scale value and keeping Blockchain relevant in 2017. With more than 20 consortia in place already, we are on our way to success.*" (Young, 2017).

Likewise, the involvement of participants in blockchain development project in addition to early involvement of regulators and legal entities is accentuated (Buehler et al., 2015; Gabison, 2016; Walport, 2016). It is critical that Fintechs and financial institutions collaborate instead of competing with each other (Belinky, Rennick, & Veitch, 2015) in order to create standards for digitizing assets and value transfer between banks. The payoff for cooperation may be industry utilities and faster development cycles (Buehler et al., 2015). To this end, there are around 20 consortia around the world. Over 70 banks around the world have joined in one consortium called R3 CEV and it is the largest in the world. R3 was originally designed to investigate and roll out distributed ledger technology to global banking, leveraging Blockchain technology in a private, centralized environment to harmonize and add efficiency to a range of internal and external processes (Gendal Brown et al., 2016). IBM and Microsoft are already providing distributed ledger in their cloud services and Linux Foundation has launched an open source blockchain project called "Hyper Ledger" built upon Ethereum platform (Linux Foundation, 2015). Each blockchain solution provides different consensus mechanisms, encryption algorithms, and/or auditing permissions. Authority or voting weight can depend on factors including a proven stake, trust in a central validator or demonstrated computational power(Tschorsch & Scheuermann, 2015).

Thomson Reuters said that market participants understand the potential, though they are still trying to work out whether blockchain can offer a cost-cutting opportunity or represents a margin-eroding threat to certain areas of their businesses (Andrews & Manuel, 2016). Wences Casares, co-founder and CEO of Bitcoin wallet and vault service Xapo, acknowledged the role and impact of Bitcoin and cryptocurrency on the market. On the other hands, he was skeptical whether blockchain, on its own, will have much of an impact. He described blockchain as shared database "*which have long existed*" but the only difference is that it is truly open and free for anybody to join, unlike shared systems that organizations have used in the past. It holds participants accountable because it publicly tracks all user's activities (Kokalitcheva, 2015). In fact, most literature about blockchain was reluctant to consider Bitcoin protocols and its principles for legal institutions. Lionel Laurent stated that common element in recent initiatives do not look like blockchains in the purest sense. They are more like regulator friendly shared databases with centralized authorities and pre-approved users, rather than an egalitarian network of peers who publicly record and verify trades (Laurent, 2017). In an analogy, (Shaw, 2016) described the difference between bitcoin blockchain and DLTs as the difference between the internet and intranet, in terms of capability and impact. This is derived from the fact that until today, the only successful implementation of blockchain is Bitcoin and Ethereum. Both of them use blockchain in its original form. On the other hands, despite the large investments in DLTs, until now there has been no commercial implementation introduced. Depository Trust & Clearing Corporation, the New York-based entity that settles and clears U.S. stock and bond trades, is building a distributed ledger for post-trade processing of derivatives. It seems very much like a centrally managed DTCC system and there has no indication of cost savings. But its aims are at least clear: record and manage agreements in the cloud without error and free back-office staff from grunt work (Laurent, 2017).

From legal point of view, policy makers acknowledged the disruptive characteristic of the technology and embraced the advantages blockchain offers (Custers, 2015). The recorded information on the distributed ledgers can go beyond currency and transactions. Some recent projects around the globe look to provide services that are traditionally provided by public entities. In Estonia, the government has launched a project with a private company called Bitnation to provide e-residency and notarization services through a blockchain (Allison, 2016).

Furthermore, among the potential applications, the UK Chief Scientific Adviser discusses how to apply blockchain to pensions, aid and general governmental expenditures (Walport, 2016). The governmental entities can write on a public ledger all their expenses and the ledger could be available for all to see. These measures would encourage transparency and accountability and is a powerful tool to prevent corruption (Allison, 2016). This might be viable for public sector and non-for-profit organizations. However, privacy is a major concern for regular businesses and placing their data on a public blockchain is surely unattractive option. Even with consortium blockchain, it is unclear which data participants can share and which cannot many authors did not consider this aspect in their publications.

Moreover, Nitin Gaur, a director at IBM blockchain labs, suggested blockchain can be an added value for enterprises but emphasized that it must be monitored to satisfy regulations and generally accepted IT practices for purposes of high availability, capacity planning, fault identification and pattern recognition (Nitin Gaur, 2016).

In summary, the blockchain provides immutable data storage, which only allows adding transactions without updating or deleting any existing transaction on the blockchain to prevent tampering and revision. The whole network reaches a consensus before a transaction is included into the immutable data storage. Based on these attributes, authors have shed light on promising use cases beyond value transfer, but most literature lacks details about implementations. Publications are mostly enthusiastic about blockchain and describe it as revolution. They expect unlimited use cases to emerge. However, others believe that the application of blockchain rather than currency transfer would be limited since the only successful implementation is bitcoin until today. In terms of using digital currency in private blockchains, the reader can spot a contradiction about using digital currency in private blockchains; digital currencies are volatile and authors agree that they do not fit businesses transactions. However, many use cases suggest using Ethereum platform, which uses Ether in representing assets and cost fees for every transactions. Overall, our literature review revealed a gap in the knowledge base about different aspects including privacy, performance, and the impact on different value chains.

Chapter 3
Environment Analysis

Environment analysis includes the business environment, both processes and technology architecture. This chapter will present analysis of business problems and opportunities in addition to comprehensive technological analysis of enterprise systems, their functional and technical infrastructure in addition to the advantages and challenges.

3.1 Enterprise Systems (ES)

An enterprise system is a business management system that comprises integrated sets of comprehensive software, which can be used to manage and integrate all the business functions within an organization with a rationalized data architecture characterized by core process integration and shared product and/or customer databases (Ross, Weill, & Robertson, 2006). It gives organizations and companies an incorporated real-time view of its core business processes such as production, planning, manufacturing, inventory management, and development (Bahssas, AlBar, & Hoque, 2015). In addition, it allows automation and information sharing to reach best practices in managing business process and have different modules that run variety business activities such as accounting, finance, supply chain, human resources, customer information, and many others. ES makes it possible for organizations around the world to response to new customer needs and reach higher market opportunities. It improves quality, customer satisfaction, performance and profit (Bahssas et al., 2015).

Enterprise systems were used since 1960's and at the beginning; companies wanted a solution to control their inventory by using Inventory Control packages (IC) (Bahssas et al., 2015). Then Material Requirement Planning (MRP) was developed in the 1970's. It is a calculation technique for planning purchase orders and manufacturing orders (Ng, 2002). The system then was extended in the late

1980's to MRP II to emphasis optimized manufacturing processes to form a character based ERP. MRP II was a systematic approach to plan production in complex multistage manufacturing systems. The major inputs of MRP II were the consolidation in data between productions, inventory status, demand management, forecasts, and purchasing (Ng, 2002). In 1990 ERP system were developed as Client/Server to integrate business processes such as manufacturing, distributions, accounting, finance, human resources, inventory management, and project management (Bahssas et al., 2015). In 2000, ERP extended to focus on integration, transformation, and collaboration in the business value chain and a new concept was proposed to be a web based environment, which made the way for Services Oriented Architecture (SOA) to be developed and that allowed different systems to communicate with each other. SOA become the standard that ERP vendors work toward (Bahssas et al., 2015). Later on, the emergence of cloud computing made it possible for ERP systems to be provided as a service on the cloud (SaaS). SaaS simplified the adoption of ERP and the extensibility of existing deployments. This new ERP deployment model will require less hardware investments, as well as less fees and internal hazard for system maintenance and upgrade. These attractive cloud features therefore result in an increasing trend for companies to consider migrating their internal ERP applications and databases into the cloud (Peng & Gala, 2014).

Cloud ERP brings many advantages to manufacturing firms, including increased data storage, real-time data access, scalability, and reduced costs. Cloud ERP lets firms access data in the cloud on any device from anywhere in the world in real time, and then make critical business decisions on the fly collaboratively. This leads to greater efficiencies, greater productivity, and dramatic increases in the ability to react swiftly to changing customer demands and greater profits (Garrehy, 2016).

"In postmodern ERP, ERP is neither defined as a single integrated suite, nor is it a specific set of modules. It is no longer a "thing." Instead, each organization must define its own ERP strategy in terms of the administrative and operational capabilities it encompasses, which applications will be used to support these business capabilities, and how they will be integrated (which may range from tightly integrated to very loosely coupled).” (Garrehy, 2016)

According to Gartner, as large on-premise ERP systems quickly become obsolete, companies are looking for a decentralized, hybrid system of mixed vendors, mixed functionality and in some cases, an even split of on-site and online systems. Gartner believes that businesses can take advantage of the lower costs, better functionality, and increased flexibility provided by mixing cloud applications with on-premise applications.

3.2 Enterprise System Architecture

3.2.1 ES Functional Architecture

There are many ERP modules in ERP software solution, where module corresponds to a major functional area of an organization. Modularity in ERP systems allows components to be pluggable, which means that organizations can choose to implement modules that fit with their business needs. In general, any ERP includes the components mentioned below (Motiwalla & Thompson, 2012):

- Production Module: previously known as Manufacturing Requirements Planning (MRPII), this module assists in planning and optimizing the manufacturing function by using historical data to project production and sales and determine what resources will be needed at a given time.
- Purchasing Module: this module assists the purchasing functions of an organization by identifying existing and potential suppliers of raw materials and suppliers, negotiating prices, awarding sales to suppliers, and billing. This module is often integrated with supply chain management software and business-to-business applications so that orders with suppliers can take place electronically and with limited intervention.
- Inventory Management Module: this module supports the inventory management functions of an organization B managing appropriate levels of materials

and supplies based off projected needs. Inventory is monitored, reconciled, replenished, and reported on by the module so that inventory is on target and not over-stocked or under-stocked.

- Sales and Marketing Module: the sales portion of this module supports the revenue generating functions of an organization through entry, scheduling, shipping, and invoicing of customer orders. This function is often front-ended through e-commerce websites and online stores. The marketing portion of this module supports the generation of sales leads and advertising.
- Human Resource Module: this module manages all of the human resource information of an organization, such as employee demographics, salaries, benefits, performance evaluation, and promotions. Often this module is integrated with Knowledge Management Systems (KMS) that can compile employee data and allows an organization to best utilize the skills of each employee.
- Finance Module: this module compiles financial data from the other functional modules of the ERP system and generates reports that describe the financial position of an organization. Reports include budgets, balance sheets, general ledger, trial balance, and financial statements.

Due to integrated nature of functioning, a few master tables are referenced frequently all across the system and databases, and shared by different applications, functional areas, and sites. Data incorporated thereon need to be accurate, complete, timely, and consistent. The quality of data in master tables, is a major reason for success or otherwise of an ERP system (MSG.come, n.d.).

3.2.2 ERP Technical Architecture

Traditional N-tier architecture: N-tier ERP architecture was introduced as a solution for problems that existed in the previous Two-Tier design. It consists of three layers (presentation, application, and data layer). The presentation layer is where Graphical User Interface (GUI) presents the data for end users. Application layer is responsible for distributing requests across different applications servers and for business logic execution (Bahssas et al., 2015). The application layer facilitates communication between data layer and user interface. Data layer is responsible for data storage and management and is actually one or more databases. (Figure 3-1) shows the structural design for a traditional ERP system.

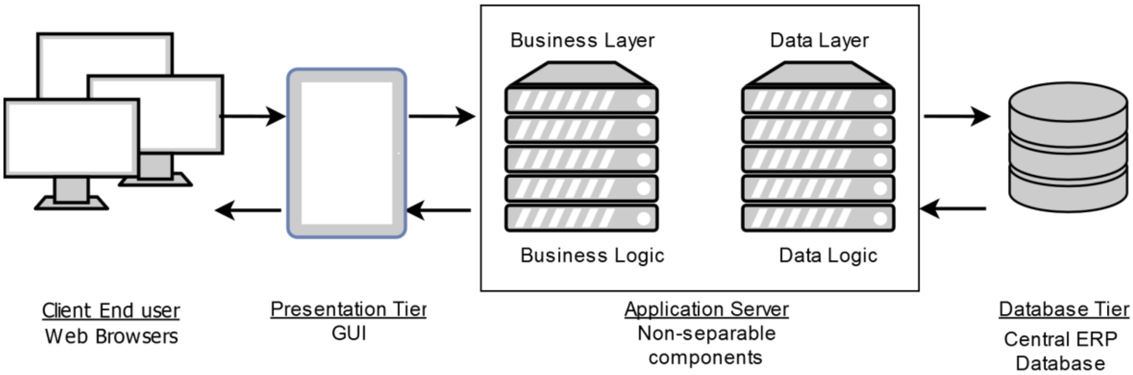

Figure 3-1 N-tier ERP architecture- source (Bahssas et al., 2015)

Web-based architecture: This architecture was conceptualized by (Ng, 2002) and then elaborated on by (Tarantilis, Kiranoudis, & Theodorakopoulos, 2008). The design development was facilitated by the improvement in internet technology. It consists of three layers (user interface, web server, and application/data server). Web server is responsible for HTML, and ASP transactions and act as a gateway between Application/data server and user interface (web browser). Web application/ data server is responsible for application distribution and database storage (Tarantilis et al., 2008). (Figure 3-2) shows the structural design for a Web-based ERP system.

Web-based architecture enabled a web browser to access multiple distant applications and databases. A middle-tier communication software layer is responsible for extracting and sending parameters through HTTP protocol to the remote application and formatting the results into webpages (Bahssas et al., 2015).

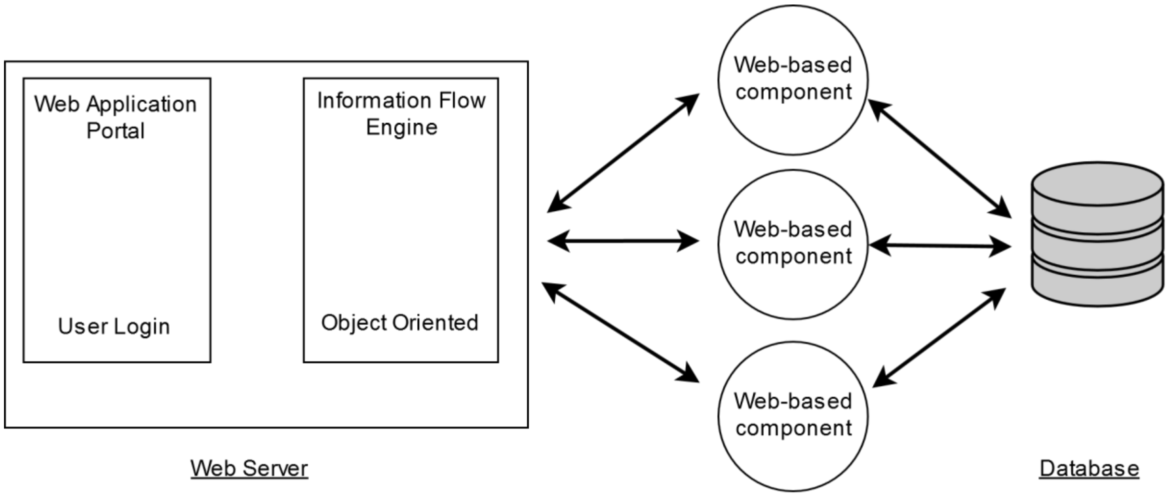

Figure 3-2 Web-based ERP Architecture

Web-based architecture introduced a new Web-Based Object-Oriented Model (WOOM). This technology allows system-to-system integration that facilitates processes between multiple systems. Further, this model allowed modularity in the system, which means that the system does not have to contain all modules to operate and the system can be customized according to enterprise' needs.

In addition, a Web-based architecture reaches higher performance because of web server layer that integrate ERP applications with existing systems and web browser. This architecture design makes faster client/server response, improves data integrity, allows for easier modifications, and has a higher flexibility (Bahssas et al., 2015). Web services enable seamless data access to the authenticated users at the right time from everywhere without the need for specific software clients. According to (Tarantilis et al., 2008), integration can be achieved with superior reliability, security, manageability, testing and effectiveness. Web Services use object-oriented technology to mix data and programming elements in Web Service methods that can be accessed by different applications. Web Services enable proprietary applications to communicate over the Web.

3.3 SOA and ESB

As organizations grow, they expand their IT infrastructure in response to their growth. A typical scenario is one enterprise runs considerable number of applications, which can be a mix of legacy on premise, web applications, and many individual solutions that are custom for each department. These applications should be able to communicate with each other in order to support business processes and achieve competitive advantages. This diversity in enterprise system landscape is, on the one hand advantageous in a way that it enables IT managers to select solutions that are the best for their particular purpose. On the other hand, it is necessary because according to (Menge, 2007) is nearly impossible to develop one huge application which performs all business functions of a typical enterprise due to the enormous requirements. Due to the rapid and continuous development in software systems, integration between different applications become a fundamental requirement although there are different lifecycles and lifespan for upgrade, expand, and replacement.

Early solutions to integrate systems like Remote Procedure Call (RPC) failed quickly to meet the challenges in flexibility, security, and reliability. That is because of the tight coupling and interdependency between connected applications. For example, any changes to the interfaces will need to be propagated to the code base of both systems, failure in a subsystem could cause the entire system to fail, and performance is adversely affected if a subsystem do not scale equally to other subsystems.

This led to the emergence of a new mechanism called Message Oriented Middleware (MOM) (Curry, 2004). Nevertheless, MOM had a big problem. These solutions often use proprietary protocols and platform specific interfaces and deployments. This leads to a total dependency of the applications on the infrastructure and causes interoperability problems between different vendors' islands (Menge, 2007).

Service oriented architecture (SOA), as described by (Menge, 2007), is an architecture concept which defines that applications provide their business functionality in the form of reusable services. A service in that context is a self-contained and stateless business function that is accessible through a standardized, implementation neutral interface. With this approach, complex business processes are implemented through a combination of several services as shown in (Figure 3-3).

SOA allows complex enterprise applications and end-to-end business processes to be composed from these services despite the differences in operating system platforms and programming languages. However, Falko Menge stated that the adoption of SOA requires refactoring, wrapping or replacing legacy applications with new standards-aware equivalents. Which is a costly, incremental, and slow process.

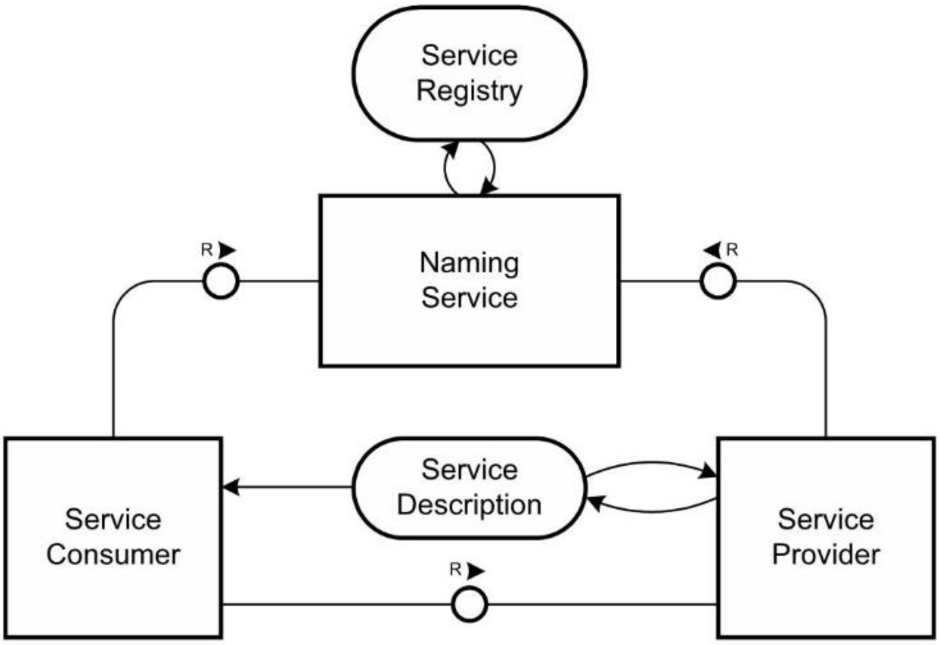

Figure 3-3 typical service oriented architecture – source (Menge, 2007)

The need for powerful integration solutions that support open standards in addition to the support of SOA led to the idea of an Enterprise Service Bus (ESB).

ESB is an open standards, message based, distributed integration infrastructure that provides routing, invocation and mediation services to facilitate the interactions of disparate distributed applications and services in a secure and reliable manner. ESBs are usually realized through service containers distributed across a networked environment. These containers host integration services like routers, transformers, application adapters, or MOM bridges and provide them with a broad range of communication facilities. In order to support SOA the ESB service containers have to include all important web service technologies (Menge, 2007). See (Figure 3-4) for a SOA-oriented integration environment using ESB.

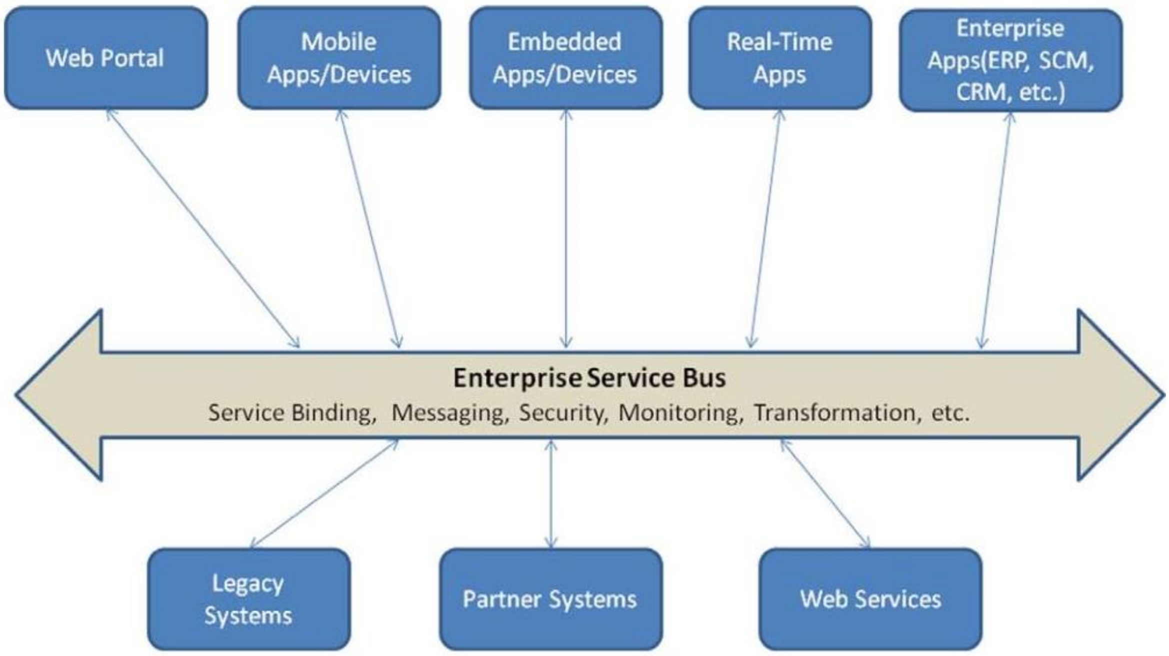

Figure 3-4 SOA-oriented integration environment using ESB – source: (He & Da Xu, 2014)

Although ESBs provide coordination of interaction of the various resources and provide transactional support, achieving successful integration requires implementing a master data management (MDM) system, which is explained in the following subsection.

3.4 Master Data Management

Most software systems have lists of data that are shared and used by several applications that make the enterprise system. A typical ERP system as a minimum will have a customer master, an Item master and an account master. This master data is one of the key assets of a company. However, the pain that organizations are experiencing when it comes to consistent reporting, regulatory compliance, or adopting SOA, is the failure to manage their master data (Wolter & Haselden, 2006).

Due to the diversity of data, it is challenging to identify elements of data that should be managed by MDM system. (Table 3:1) lists the five essential types of data in corporations.

Because it is used by multiple applications, an error in master data can cause errors in all the applications that use it. For example, data about the same customer might differ between billing, sales, and marketing systems. Even if the same data exist, differences can occur in data models in these systems.

In addition, even when an organization succeeds in achieving a single set of master data amongst its systems, companies grow through mergers and acquisitions.

Type	Source
Unstructured	Data found in Email, web portals, PDF, articles
Transactional	Data generated by interacting with databases
Metadata	Data about other data stored in XML documents, log files, configuration files. Report definition.
Hierarchical	Data about the relationships between other data. It describes real-world relationships for accounting and organizational structures.
Master	Data that are critical nouns of a business. They describe people (customer, employees, suppliers), things (product, store, asset), concepts (contract, warrantee, licenses), and places (office locations and geographic divisions). Each of these groups can be subdivided into other groups.

Table 3:1 Types of Organization's data- source: (Wolter & Haselden, 2006)

Each company comes with its own master data. Merging master lists together can be very difficult and it cannot be achieved by normal database joins. According to (Wolter & Haselden, 2006), in many cases, fundamental changes to business process will be required to maintain clean master data, and some of the most difficult MDM issues are more political than technical. The second thing to note is that MDM includes both creating and maintaining master data. Investing a lot of time, money, and effort in creating a clean, consistent set of master data is a wasted effort unless the solution includes tools and processes to keep the master data clean and consistent as it is updated and expanded.

3.5 Enterprise Collaboration capability

Collaboration is the willingness of departments to work together, having mutual understanding, having a common vision, sharing resources, and achieving collective goals.

Collaboration capability is the ability to build and manage relationships with other parties on individual, team, departmental and organizational levels in order to empower knowledge sharing, which in turn, underpins the continuous innovation and facilitates competitive advantages (Blomqvist & Levy, 2006). It consists of information processing, communication, knowledge transfer, and control, the management of coordination on various levels, trustworthiness or the ability to engender trust, and negotiation skills (Tyler 2001 cited by Blomqvist & Levy, 2006).

According to (Blomqvist & Levy, 2006), various authors agree on the critical role of trust, commitment and communication in collaboration. These three elements are the differentiators between relationally oriented relationships, which imply common goals, shared values, mutual commitments, and collaborative behavior, and transactional relationships where departments are considered independent entities competing for company resources. These two types of relationships are also applied to inter-organizational relationships. Relationships between organizations are seen to generate more value and competitive advantage. Relational benefits rise from investments in relationship-specific assets, knowledge exchange and learning, and complementary capabilities. (Figure 3-5) depicts the multi- and cross-level concept of collaboration capability.

The benefits of ERP systems, as discussed before, are to integrate various systems and working processes and leverage efficiency. The integration of a Supply Chain Management (SCM) module in the ERP is fundamental to interact with suppliers and business partners in order to obtain raw materials and resources needed to bring finished goods or services to market in a great speed, efficiency and overall quality (CompuData Inc, 2015). It is in the best interest that all parties fully understand the importance of collaboration between all stakeholders involved in the value chain.

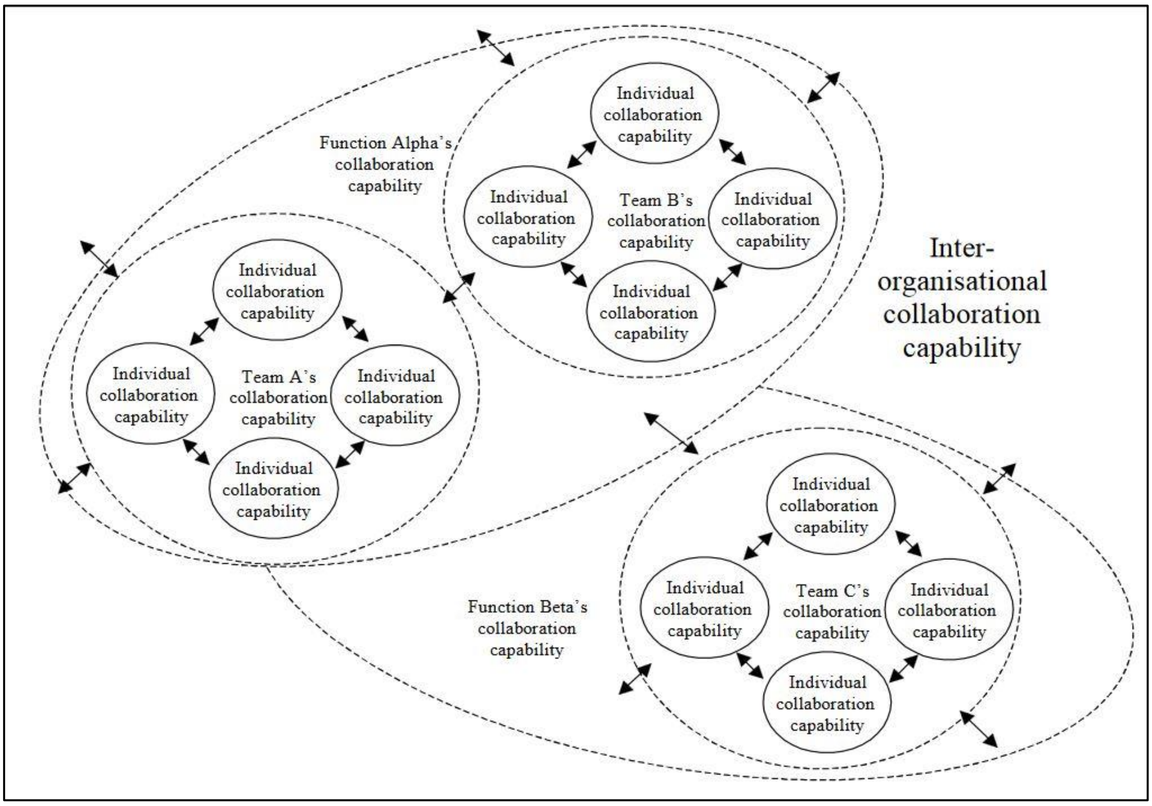

Figure 3-5 Collaboration Capability – source: (Blomqvist & Levy, 2006)

Information sharing along the value chain across departments and with business partners is regarded as a crucial factor for achieving competitive advantage. However, there are some barriers to information sharing within a supply chain. Among these barriers are confidentiality of the information shared, incentive issues, reliability, and cost of information technology. Furthermore, anti-trust regulations, accuracy of the shared information, and the development of capabilities that allow companies to utilize the shared information in an effective way (Lotfi, Mukhtar, Sahran, & Zadeh, 2013).

From a technical perspective, collaboration refers to integrating different systems in order to exchange data from one system to another with minimum manual intervention.

3.6 Enterprise System Integration

The deployment of ERP systems often requires business process reengineering in order to align business with the system (Hasselbring, 2000). This is unacceptable for some businesses. Therefore, the information architecture should align with the business. Systems Integration (SI) aims at supporting the business process, while preserving the investment in legacy systems (Hasselbring, 2000). Pre-existing applications must still be able to communicate and exchange data without modification. The speed of business and technology change does not allow time for total replacement, therefore, evolution and migration of legacy and new application systems is required. Migration and evolution aim at protecting existing investments and enabling rapid response to the changing user requirements. For managing the evolution of those complex systems, it is necessary to deal with change on the organizational level, group collaboration level, and system level in a coherent manner (cited by (Hasselbring, 2000)).

Organizations deploy different applications, which are developed independently by different vendors and at different times, this leads to heterogeneity. Heterogeneity comes from differences in hardware platforms, operating systems, database management systems, and programming languages. This also results in variety in data models. In order for these applications to interoperate, data integration is required.

Data integration involves data mapping and conversion and mapping relationship between data source schema and target data schema (He & Da Xu, 2014). Middleware and web services facilitate data integration. However, different middleware has different advantages; many enterprises have used various middleware over the years. According to (Baker & Dobson, 2005) middleware technologies and products from different vendors often have trouble to easily interoperate. There are different tools that support integration and typically, some large vendors provide their own custom middleware as part of their solution, which allows organizations to choose between different strategies for integration. (Table 3:2) lists the options for integration:

Integration strategy	Description
Real-time data replication	In real time, synchronize the data in one system with the data in another system.
Batch data replication	At certain times, synchronize the data in one system with the data in another system.

Data sharing	From system A, access and change data in system B. do not store the data in system A.
Presentation layer integration	From application A, access screens from application B.

Table 3:2 Integration strategies – source: (Oracle Inc, 2013)

These integration scenarios are possible to apply when different systems are provided by the same vendor and/or a comprehensive data management system is in place, as mentioned before. Yet, collaboration and knowledge sharing pose additional challenges in situations where companies focus on their core-business and outsource their sub-processes. The number of businesses that collaborate in the value chain increases; each has their own system and data models, which increase complexity of the integration process. Within an ecosystem where one enterprise deals with different suppliers and outsourcers, difficulties in achieving collaboration and trust between trading parties lead to the emergence of different kinds of third parties to facilitate this collaboration.

To sum up, Enterprise System makes it possible for organizations to respond to new customer needs and reach higher market opportunities. It improves quality, customer satisfaction, performance, and profit. ESs have been in continuous improvement since the 1960's and major diversion points are related to developments in networking technologies. Organizations strive for competitive advantage through adoption or updating technology. Therefore, they adopt different applications to support their business operations but at the same time, try to preserve their investment in legacy systems. This has led to diversity in the overall ICT architecture, which made it necessary to provide techniques to allow these different systems to communicate together. Solutions to integration issues are mainly using Service Oriented Architecture and Enterprise Service Bus to facilitate communication. However, this accompanies challenges caused mainly by the underlying data layer. It is common for the different techniques to cause a different execution method within the database. This is depended on how the instance, the database, and the data structures within database are configured. To overcome these issues, Master Data Management system is necessary, but this requires data migration and remodeling for all applications, which are costly and time consuming operations.

Chapter 4
Results and Findings

Our design process started with initial interviews with ERP experts in order to identify potential areas for blockchain to be a benefit within enterprise systems. The interviews focused on characteristics of blockchain solutions compared to those provided by existing systems. The results of these interviews are presented in the first subsection of this chapter, which forms the basis for the proposed artifact in the following sections. In order to develop an effective solution, a preliminary design was made based on the knowledge base and interviewees' remarks. Afterward, design validation took place in order to evaluate the proposed artifact and identify areas of improvement.

4.1 Expert Interviews

There were two sets of interviews. The preliminary interviews aimed at exploring the possibility for blockchain within the enterprise world and gaining insight into business processes and real-world challenges in the existing enterprise systems. In particular, the focus was on concepts related to blockchain technology such as sharing a single source of data, process automation, forcing business logic, security, trust, and third party involvement. Since participants from the ERP domain did not have clear idea about blockchain, it was necessary to give a short explanation about its basic concepts.

When ERP consultants were told that blockchain is a shared database that provides "a single version of truth" in the real time across participants and it supports smart contracts to automate processes and force business logic, their reaction was that ERP systems are aimed to integrate the management of all major business processes in an enterprise. Packet Based Solutions (PBS) have at their core a common database that provides "a single version of truth" in real time across multiple departments and functions in different locations. They also support sophisticated business processes though database stored procedures and triggers, which are far more advance than smart contracts. In terms of robustness, traditional shared databases use different strategies to improve sustainability and throughput, and reduce latency such as master-slave and multi-master replication. Therefore, we looked for the advantages of blockchain over the existing systems.

Nevertheless, further discussion about ERP implementations revealed some interesting facts. There are two categories of PBSs; one is comprehensive solution that aims at surrounding the whole operations of an enterprise such as ORACLE and SAP. These solutions provide various modules that can support organizations in different industries. However, these solutions are expensive and although they are highly configurable, they still comprise unnecessary service for which clients must pay. Therefore, mostly large-sized organizations implement such solutions. On the other hands, small and medium organizations opt for less comprehensive, cheaper solutions such as NETSUIT, UNIT4, or EXACT. These solutions comprise a limited number of basic ERP modules, which in some cases cause those companies to buy additional applications to support their specific operations. According to the functional consultant, implementing custom applications is common within organizations of any size.

"*Diversity in the ICT architecture of an enterprise is inevitable. My job as a consultant is to provide advice about the best solution or set of solutions to support operations and justify IT investment*" L. V. D.

Diversity in ICT architecture is a result of fast changing business environment, merger, acquisition, outsourcing, and business processes reengineering. All aforementioned factors make it challenging, if not impossible, to have a single operational database[2] upon which, all enterprise applications are running. This is also confirmed in the literature (Hasselbring, 2000; He & Da Xu, 2014).

In order to get an idea about how this diversity has effect on businesses, we went step-by-step through some business processes, in particular, processes that require interaction between different parties. We could identify some inefficiencies and

[2] Operational databases used by applications and updated by business daily operations.

third party involvement caused by the inability of different systems to communicate with each other. Inefficiencies represented by manual checks and email communications between functions because each of them cannot have access to the database of the other's. In addition, third parties are involved in processes that include business partners especially to facilitate trust and records reconciliation (one of these processes will be explained later in this chapter). Furthermore, the historical evolution of an enterprise system (legacy systems) has led to inherently different business practices and information concepts. These differences are attributed in the first place to different data representation amongst software vendors, organization operations, and the type of items/services processed.

When asked whether there are solutions or measures in place for addressing these issues, the interviewee said:

"To avoid these differences, there are Universal Business Language (UBL) systems introduced to unify data sharing within organizations and beyond organization border with business partners and concerned authorities, but implementing such a project is a major data integration process and requires involvement and collaboration from different parties. The municipality of '…', for example, hired us to implement such a system but did not carry on because they deal with too many suppliers, each of which uses different accounting system. They exchange information by email and for invoicing they are using a third party."

UBL is an open, customizable, and extensible XML vocabulary of business documents, which are exchanged between trading partners such as buyers, sellers, shipper, and warehouses. There are different standard vocabularies for different types of document family. UBL was introduced in 2003 but it became recognized as an international standard by 2015 (Ken Holman, 2017). (Figure 4-1) shows how UBL software works in combination with an application.

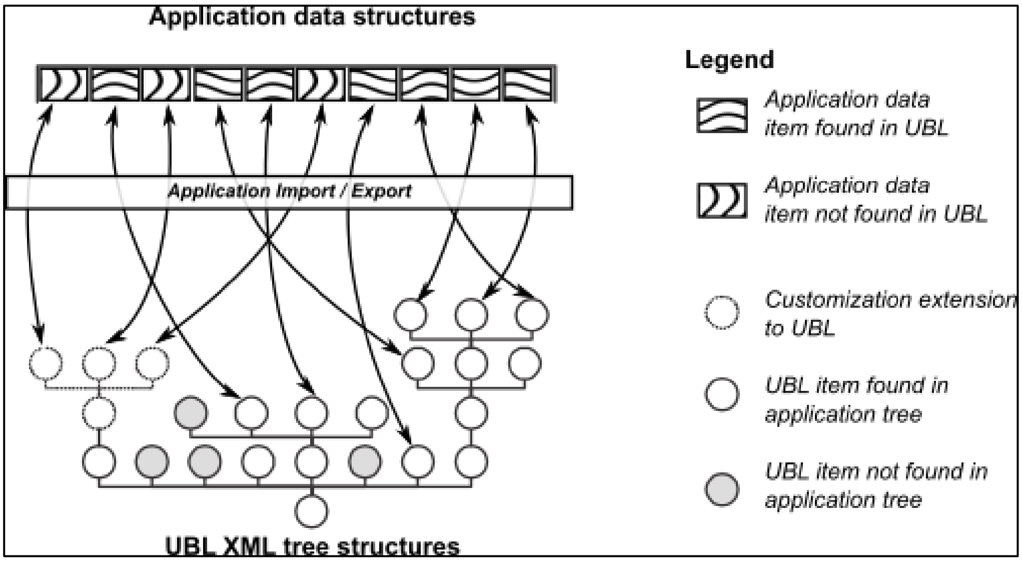

Figure 4-1 Universal Business Language System- source (Ken Holman, 2017)

Similar extensions need to be installed on sender and receiver's systems. However, what is happening in practice is that trading partners hire a third party "business documents provider", to match their documents, which in turn implement a system to change the message format into the one used by the other party. This is shown in (Figure 4-2).

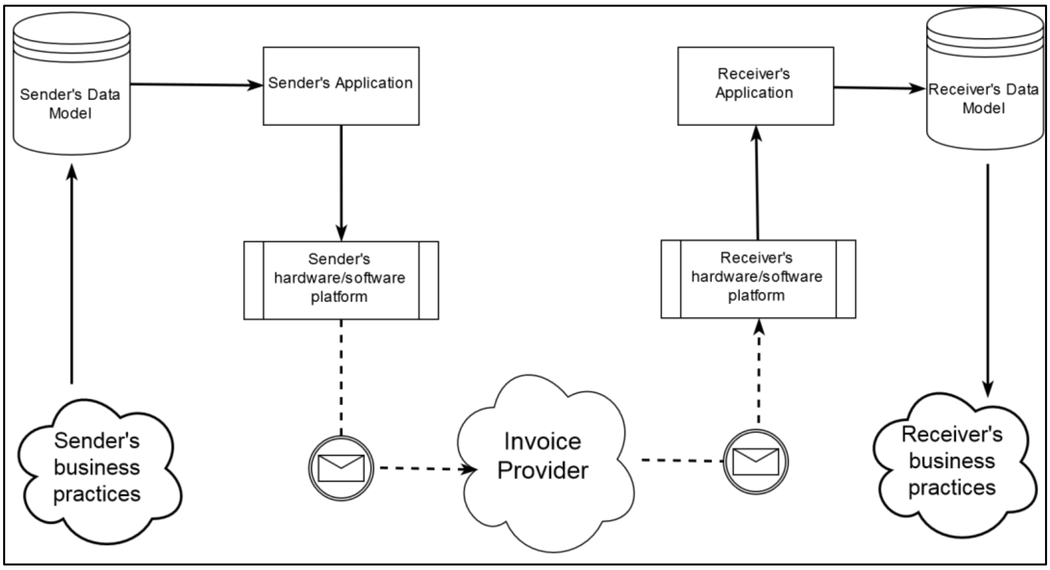

Figure 4-2 Invoice matching using Invoice Provider

In order to gain insight into technical challenges that resulted in communication and interoperability issues, it was necessary to interview an integration expert.

We know from the literature that ESB using SOA (subsection 3.3) is an architectural solution to tackle the problem of enterprise silos and support interoperability and connectivity between different systems. However, the integration expert stated that an ESB should be conceptually able to facilitate communication between systems in a pattern that looks like human communications. Nevertheless, IT systems are inflexible in their way of representing concepts. For instance, people from different functional teams such as marketing, sales and accounting can send emails to update each other on status and they can understand each other. On the other hands, their systems cannot communicate in the same flexible way: the client data model must be identical in all three systems in order for them to communicate. Therefore, each integration project, either between two on-premise systems or with cloud, must include a Mater Data Management system (subsection 3.4).

In his interview, the Integration Consultant mentioned several social and political factors involved when it comes to standardizing data among departments or standardizing processes among business partners. However, apart from organizational, behavioral, and political challenges, some of the technical challenges in integration, extracted from expert interviews, are listed in (Table 4:1).

An additional interview was with a blockchain consultant who confirmed the technical challenges in terms of performance when implementing blockchain on large scale. He also added that implementing blockchain as a database for operational applications is not a favorable solution since they are immutable. In addition, querying the ledger is much more difficult than fetching data from a relational database. He stated; "the ledger is basically a sequential database where every transaction creates a new record, which means that the size of the database will grow exponentially if it is used for operations. Think of the old sequential database and why they were replaced by relational database" (J. W.).

Challenge	Example	Reason
Keep data in sync	Naming formats	Integration fails if names not formatted equally
Re-use interfaces	Extending with additional columns	Requires rework on existing integrations
Combine data from different systems	Error handling if not all additional data is found	Missing records leads to no full synchronization.
Use unique identifiers	Some interfaces require also additional data to identify relationships.	Will only work if additional data is completely synchronized.
Enrich data for target system	Required data for target system not available in source system	Requires definition of default values – to be added somewhere in the flow.

Table 4:1 Technical Integration Challenges

He also added that existing blockchain solutions are standalone and most of them use digital currency, which limit their use cases and hinder adoption.

As a result, the preliminary interviews revealed some challenges within enterprise systems that the proposed artifact should solve. The following subsections will propose a solution bearing in mind the technical capabilities and limitations of blockchain, the existing infrastructure, and business environment.

4.2 System design

In the proposed design, we elaborate on the principle of data independence, which is the separation of implementation choices from the database conceptual model. This separation is also required in designing a blockchain platform since the purpose of the implementation is to implement a blockchain in the current system instead of adapting the system to a blockchain. The data independence principle is also followed in the development of enterprise systems that use a central database for their operations. In this, we are referring to (Den Haan, Albani, & Dietz, 2009) who explained the enterprise ontology theory and emphasized the focus on the construction and operation of a system rather than the functional behavior. According to Den Haan, this theory can offer advantages in understanding the essence of an organization and in using organization models as a starting point for building software supporting organizations. This will also support choosing the most effective level of abstraction during information system development in order to establish a clear separation of concerns. The three levels of concerns are Data, Information, and Business level.

The data level is where the artifact is technically explained. This involves explanations about different components of the system, which includes transactions, encryption, and communication. The information level is an abstraction level where we seek to provide an understanding of how the system works and how it can be used by end users. Finally, the business level represents the business implications from implementing the artifact. These three levels are explained in this chapter. We will start with a technical level explanation all the way up to the business processes.

The architecture we are going to propose in this paper presents a mechanism to implement a blockchain platform in current systems. Since ERP solutions are proprietary packaged software, adapting the architecture of the ERP system itself to contain a blockchain would be the responsibility of ERP vendors. Furthermore, implementing blockchain in a currently up and running system is not a feasible solution since it is a major overhaul project. Therefore, the best solution is to design an architecture that uses blockchain as separate software that can be connected to the existing system by software connectors. This technique is already used by different ERP providers in the market to enable the Enterprise Systems of their clients to communicate with external applications over the web. This improves scalability and interoperability between different systems within one organization or between different entities.

4.2.1 Enterprise Software Connector

The resource adapter serves as a protocol adapter that allows any arbitrary EIS communication protocol to be used for connectivity. An application server vendor extends its system once to support the J2EE Connector Architecture. This assures seamless connectivity to multiple EISs. Likewise, an EIS vendor provides one standard resource adapter that can plug in to any application server that supports the J2EE Connector Architecture.

We will take for instance a software connector provided by Java called the Resource Adapter; (Figure 4-3) shows a generic architecture for a bi-directional (inbound and outbound communication) resource adapter to and from an Enterprise Information System (EIS), provided by Java Enterprise Edition (J2EE) to connect enterprise information systems with external applications.

This architecture assumes that both the Enterprise Information System (EIS) and the external application are web-based and connect to a web server, which in turn, contains components for outbound connection operations as follow (Oracle Inc, n.d.):

- An application component that is used by the external application to submit outbound requests to the EIS through resource adapter.
- One or more connection pools within the Web server connector container for Managed Connections Factory (MCF).
- Multiple managed connections (MC1, MCn), which are objects representing the outbound physical connections from the resource adapter to the EIS.
- Connection handles (C-Handle) returned to the application component from the connection factory and used by the application component for communicating with the EIS.

The following components are used for inbound connection operations:

- External message sources (MS), which send messages inbound to WebLogic server.

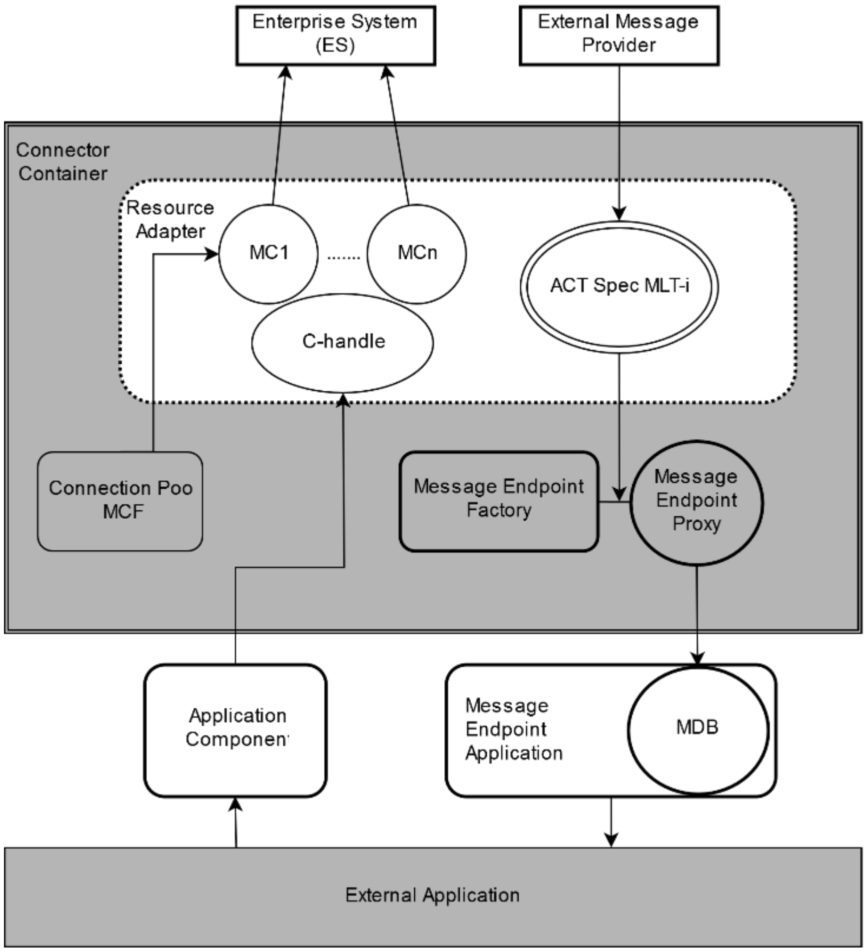

Figure 4-3 J2EE Software Connector - Source (Oracle Inc, n.d.)

- Activation Specs (Act Spec), each of which corresponds to a single Message Listener type (MLT-i).
- A message-Endpoint factory created by J2EE and used by the resource adapter to create proxies to message-Endpoint instances (a Message-Driven Bean MDB instance from the MDB pool).
- A message endpoint application that receives and handles inbound messages from the EIS through the resource adapter.

This was an example of a software connector by Oracle using Java technolgoy. Other vendors provide their own software connector such as Microsoft Navision and SAP Hanna. However, they all provide the needed functionality, connecting EIS with other applications. Therefore, differences in architecture is not an issue as long as the purpose is to achieve interoperability.

4.2.2 Software Connector to Blockchain

In the previous subsection, we explained how an EIS could communicate with other applications. In this case, the system should communicate efficiently with a blockchain platform, which has some different requirements. Therefore, in this subsection we will propose an architecture of a software connector that will facilitate communication channels with enterprise systems. The connector will comprise of components inspired from the aforementioned connector in addition to components necessary for the new blockchain functionality.

We elaborate on the software connector taxonomy provided by (Mehta, Medvidovic, & Phadke, 2000), where they stated that software connectors are the fundamental building blocks of software interactions. A connector is an interaction mechanism for the components. All software connectors comprise one or more ducts, interaction channels with no associated behavior. Furthermore, all connectors, regardless of their complexity, provide mechanisms for transferring data and/or control along a duct (Mehta et al., 2000). Connectors in distributed systems are the key elements to achieve system properties, such as performance, scalability, reliability, and security.

According to (Mehta et al., 2000), the services provided by a software connector could be classified into four categories: communication, coordination, conversion and facilitation.

Communication services transfer data among components while coordination transfers control among components. Conversion services adjust the interactions to allow components that have not been exactly tailored for each other to establish interactions. Facilitation services help to support and optimize components' interactions.

Based on the mentioned above, (Figure 4-4) shows the proposed architecture of the software connector to be used in connecting enterprise' separated applications with a blockchain platform.

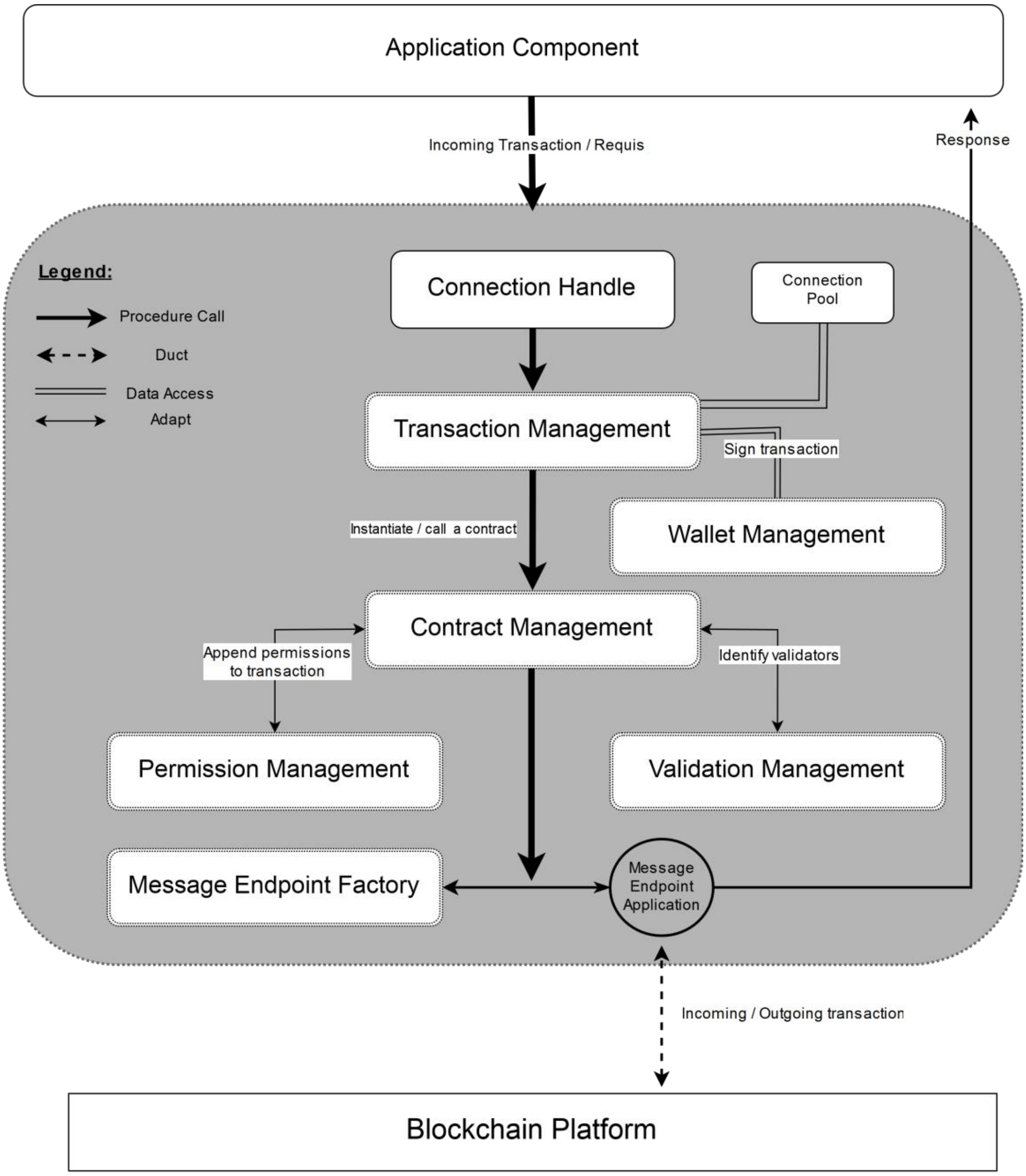

Figure 4-4 System connector for Blockchain

Service	Type	Functionality
Connection Handle	Facilitation	Access list, Identification
Transaction management	Communication	Delivery, buffering, synchronization.
Permission management	Coordination	Authorization, arbitration
Wallet	Facilitation	Authentication, signing, identity

management		
Contract management	Coordination, conversion	Commitment, automation
Validation management	Coordination, facilitation	Access list, encryption, privacy

Table 4:2 List of the components in blockchain's connector

Connection Handle: carries the connection request and user identity from the application component to enable the external application to communication with the blockchain platform.

Transaction management is a communication service and it is responsible for transferring data in a form of transaction between user's application and the distributed ledger. If this component does not buffer the data, it attempts to deliver the data at most once, if the recipient is unable to receive the data for some reason, the data will be lost. Therefore, this component must be able to buffer the data in order to be able to prevent data loss resulted from potential delay in the distributed ledger. Data is stored on the DL by adding data to the transaction, like the case with Bitcoin-like blockchain, or by sending transactions to the storage of a contract, such as Ethereum-like blockchain.

The main concepts of this domain are wallet, transaction, and node. Whereby, transactions in the system, unlike simple transaction in a Bitcoin network, are required to contain executable smart contract code in order to govern interactions with the blockchain through nodes that use their wallets to sign transactions.

From an information perspective, a transaction is not described as a block of data, but rather it is a transfer of a value object and blockchain is described according to its main characteristic as a "distributed ledger" (DL). The ledger consists of accounts, which can be of any type other than currency, balance, or equity. Whereas the business point of view, is concerned with what is created directly or indirectly by transactions. In other words, it describes the future situation after transactions are or will be applied. The word “transaction” in the proposed architecture is generic for the business processes flow. It can mean produce, use, consume, give or take of resources.

Permission management is a coordination service that manages user’s permissions to read, write, and validate transactions. It works in collaboration with all other services in the connector in addition to its role in identifying an arbitrator in case of conflict occurrence in the validation process. After transactions are included in the DL, they will be accessible for network participants who have read permissions on the DL.

Wallet management is a facilitation service that provide system users with the ability to sign transactions to write and verify signatures to read or validate. This service works in the same way as crypto currency wallet works for maintaining private keys and generating public key.

The advantage here is the double authentication. Users are already authenticated when using the external application that communicate with blockchain. In this situation, the system will handle signing the transaction automatically with minimum manual intervention. According to one of the interviewees, public key authentication is not enough to allow access to organization’s data. Therefore, users will be able to access blockchain through enterprise application where they are already authenticated.

Contract management service is the most important coordination service for maintaining contracts. The flow of all processes is translated (converted) into a code. They act like self-organizing agents during system execution. Some are executed autonomously triggered by an event, while others are instantiated by external users. In general, every contract has a storage location to write to. This storage is accessible only by the contract itself and it stores the state of the system after applying the contract. The data stored in the contract’s storage can be updated by sending transactions to the corresponding contract with new value. Likewise, functions saved in a contract can be run by sending transactions with function’s parameters to the contract. In Ethereum, for example, the state of contract storage can be queried through API. The code of contracts can also be created by submitting a transaction with the source code of the contract.

In addition, business commitment can be represented in smart contracts where transactions can be part of a contract. The commitments in the system are self-fulfilling; the committed transactions are irreversibly saved on the blockchain and executed once certain conditions are met.

Validation management service is responsible for specifying the participants who are required to verify a transaction before it is recorded in the DL. This service can send notification to the participants where they need to verify it manually. In order to preserve privacy, the blockchain in the system is of the private or consortium type, which can be implemented across multiple organizations. The validation process is controlled by trusted parties and the right to read may be public for all participants or also restricted. Read/write/validate permissions are managed in the permission management component.

When a transaction is initiated by an involved node, the system generate a secret key associated with the transaction, which is used for encryption. Based on records affected by the transaction, the validation management will identify the validators. Once the involved parties are specified, a smart contract is used[3] to facilitate the validation. Smart contract uses the access-control management component to restrict access to the record, and distributes the secret key of the transaction that can be decrypted only by the participants who receive the key. The validation is done amongst the involved participants and manual user intervention is in most cases required. Every activity about the same asset will be registered as a new transaction. Therefore, all operations performed on an asset are traceable and the status of accounts can be derived.

All digital documents that have a corresponding smart contract will contain its address. Similarly, the corresponding smart contract will contain the address of the document along with the hash value of the executed transactions.

[3] If the smart contract exists in the database, it will be called. Otherwise, new one will be created.

In a private blockchain network, participants are known and trusted. The system will increase its trustworthiness especially if legal entities are to join the network such as Tax Authority, Chamber Of Commerce, or a Bank. In addition, the involvement of a governmental authority will facilitate physical assets registration off-chain and include them in transactions on-chain such as assets transfer or depreciation. The general architecture of a decentralized enterprise system will look like the one depicted in (Figure 4-5).

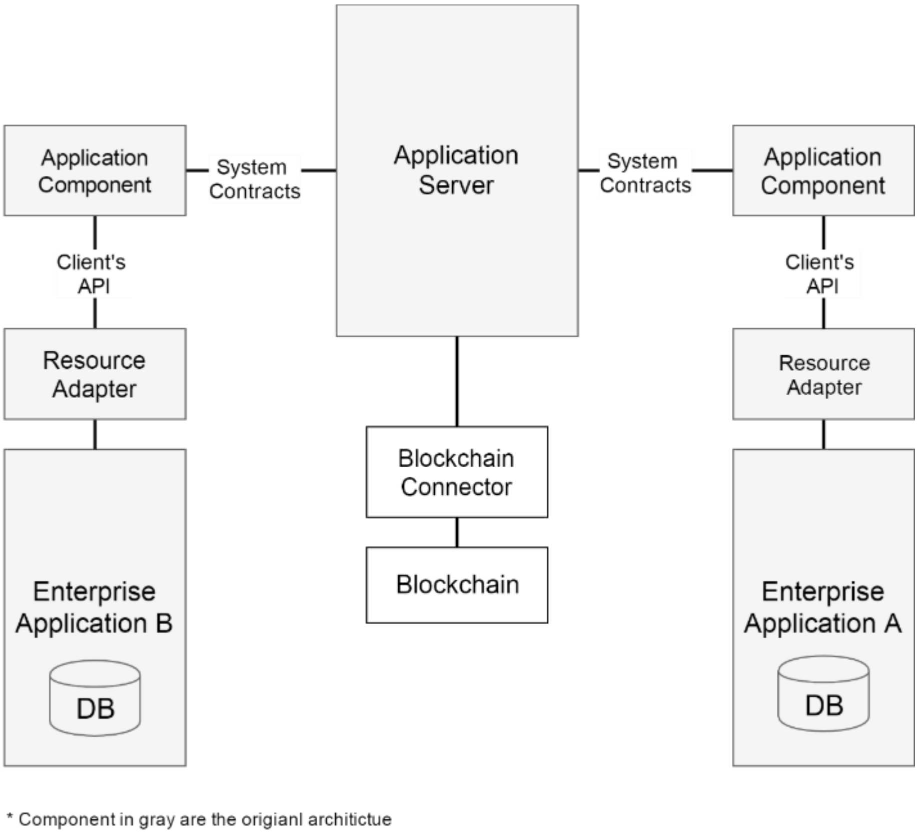

Figure 4-5 Blockchain implemented in the ES Architecture

System level contracts are implemented to achieve a standard system-level plug-ability between Web Logic Server and an EIS. The Web Logic Server implements the standard set of contract defined by the vendor-specific connector. According to (Oracle Inc, n.d.), these contracts consist of classes and interfaces that are required by the application server and the EIS, so that the two systems can work cooperatively. The EIS side of these system-level contracts are implemented in the connector classes. Similarly, we implemented components specific to a blockchain platform in our proposed connector architecture.

4.3 Design Evaluation

4.3.1 Evaluation Methods

The utility, quality, and effectiveness of a design artifact must be rigorously demonstrated via well-executed evaluation methods (Hevner & Chatterjee, 2010). (Table 4:3) list evaluation methods that can be implemented in order to evaluate IS artifacts.

Method's type	Method's approach
1. Observational	Case study: study artifact in depth in business environment
	Field study: monitor use of artifact in multiple projects
2. Analytical	Static analysis: examine structure of artifact for static qualities (e.g., complexity)
	Architecture analysis: study fit of artifact into technical IS architecture.
	Optimization: demonstrate inherent optimal properties of artifact or provide optimality bounds on artifact behavior.
	Dynamic analysis: study artifact in use for dynamic qualities (e.g., performance).
3. Experimental	Controlled experiment: study artifact in controlled environment for qualities (e.g., usability).
	Simulation: execute artifact with artificial data.
4. Testing	Functional (black box) testing: execute artifact interfaces to discover failures and identify defects.
	Structural (white box) testing: perform coverage testing of some metric (e.g., execution paths) in the artifact implementation.

5. Descriptive	Informed argument: use information from the knowledge base (e.g., relevant research) to build a convincing argument for the artifact's utility.
	Scenarios: construct detailed scenarios around the artifact to demonstrate its utility.

Table 4:3 Evaluation methods in IS design- source (Hevner & Chatterjee, 2010)

Two of these methods will be implemented to evaluate the proposed design. The first method is analytical evaluation, where we will validate the connector by analyzing its architecture using the help of the interviewees. In addition, another analytical analysis will take place to demonstrate inherent optimal properties of artifact and provide optimality bounds on artifact behavior based on expert recommendations and knowledge base.

The second method is descriptive evaluation, where we built a detailed scenario of one process to demonstrate how the system will work and support our arguments. Subsequently, the scenario will be validated by expert to check its validity and apply optimizations where needed.

4.3.2 Analytical Evaluation

4.3.2.1 Blockchain Connector validation

According to an integration consultant, a software connector should contain components that belong to the software itself in order to facilitate communication with external applications. For example, writing to a blockchain requires using a private key, which should be stored in a digital wallet. Therefore, a wallet management component was included. Similarly, contract management is necessary to store and manage system's smart contracts.

The design was then introduced to a blockchain developer who stated that in a private blockchain there should be appointed group of nodes who have writing permissions (minors / validators). However, in this case validators should not be fixed but rather appointed based on the transaction type. For this reason, the validation management component was added. Furthermore, the blockchain developer stated that organizations do not trust private/public key encryption. The encryption is a secure method for identification but it is not secure enough for authentication. In other words, a user can use his private key to identify himself and use the system but there is no way to make sure, whether it is the same one who is granted the key. In general, accessing enterprise application requires secure a connection from an authorized device via intranet. This feedback led to other additions to the connector's design. To tackle this issue, an integration expert suggested adding a connection handle and connection pool components to pass the credentials of the user, who is accessing the ledger, from the external application to blockchain connector. In addition, accessing the ledger will be allowed only via an enterprise application. This will enhance security since the same security measures are applied in order to access the ledger.

In the final iteration, the integration consultant emphasized the necessity of having a Message Endpoint Factory in the connector in order to format data for the target system, which is one of the integration's challenges (Table 4:1). He also suggested forcing a fixed format for data saved on the ledger. This will make data integration easier since external applications will have to deal with one data format, which is ledger data, instead of dealing with different data format for each application.

4.3.2.2 System optimization

In this part, we will demonstrate the inherent optimal properties of the artifact and provide optimality bounds on its behavior based on knowledge base and interviewees' recommendations. Bear in mind that the proposed architecture should be considered a supporting platform to the distributed enterprise systems. Therefore, it will work in parallel in order to facilitate data interoperability and applications integration. Such a view helps us make explicit important architectural considerations. Since the distributed ledger will not replace the operational databases, there will be no concerns about technical capabilities such as the number of transactions processed per time unit or data storage capacity. Private blockchains can support up to 3000 transaction per second (Jentzsch, 2017), which, according to the integration consultant, would be sufficient for application communications. In (Table 4:4) below some decisions are shown that should be taken by organizations who wish to implement the proposed design.

For an enterprise implementation, it is not preferred to join a public blockchain, where everyone will have access to the information saved on the chain. Therefore, organization should opt for a hybrid or private blockchain with permissioned access.

Participation	Public	X
	Hybrid	√
	Private	√
Digital currency	Token	X
	Token-less	√
Consensus	Mining	X
	Validation	√
Data scope	On-chain	√
	Off-chain	√

Table 4:4 Blockchain design considerations

In private blockchains, permissions to join the network would be granted by the owner of the chain, in this case the organization. By joining the network, any participant can, by default, read all transactions on the ledger. Nevertheless, custom permissions can be embedded in the transaction and fed to the smart contract. This includes read, execute, and validate permissions.

The system is used internally by the organization, where transactions recorded on the chain will represent knowledge transfer, business commitments, and value transfer. The blockchain consultant mentioned that using digital currency in private blockchain creates a challenge in using it for purposes other than value transfer. It is difficult to represent any transaction with tokens. In addition, each transaction will cost a fee and the mining operation causes significant delays. As a result, digital currency is completely eliminated "token-less blockchain."

Opting for a token-less blockchain means that neither mining protocols will be used: the consensus mechanism will be validation. The ERP consultant thinks that validators should be appointed based on data contained in the transaction rather than pre-appointed validators. This is similar to the validation method used in the Corda R3 platform, where validators vary according to the concerned parties.

In terms of data scope, not all data are eligible for blockchain. Bearing in mind the immutable nature of blockchain, organizations must decide on which data will be saved on the chain and which will be handled off-chain. Additionally, a blockchain-based system can maintain a unique chain to record all types of transactions together or maintain multiple chains to isolate information of separate parties or of separate concerns, for example, using one chain to store transactions, and using a separate chain to store access control information. It is also possible to use a private chain that spans across departments within the enterprise and connect that chain with a consortium chain that is shared with multiple parties in the market.

4.3.3 Descriptive Evaluation: Detailed Scenario

In order to demonstrate the artifact and its utility, we constructed a detailed scenario around it. We chose the process Purchase-to-Pay as an example of business process that requires communication between various parties. The process is described in details in (Figure 4-6).

As shown in the figure, parties from within the organization involved in the process are:

- A department that requires the materials, which might be using their own specific application,
- The procurement department, which might be using a SCM system from a vendor,
- Inventory management system, and
- Finance department that uses an accounting system.

By the vendor organization:

- Sales department that uses customer relationship management (CRM) system,
- An inventor management system, and
- The finance department.

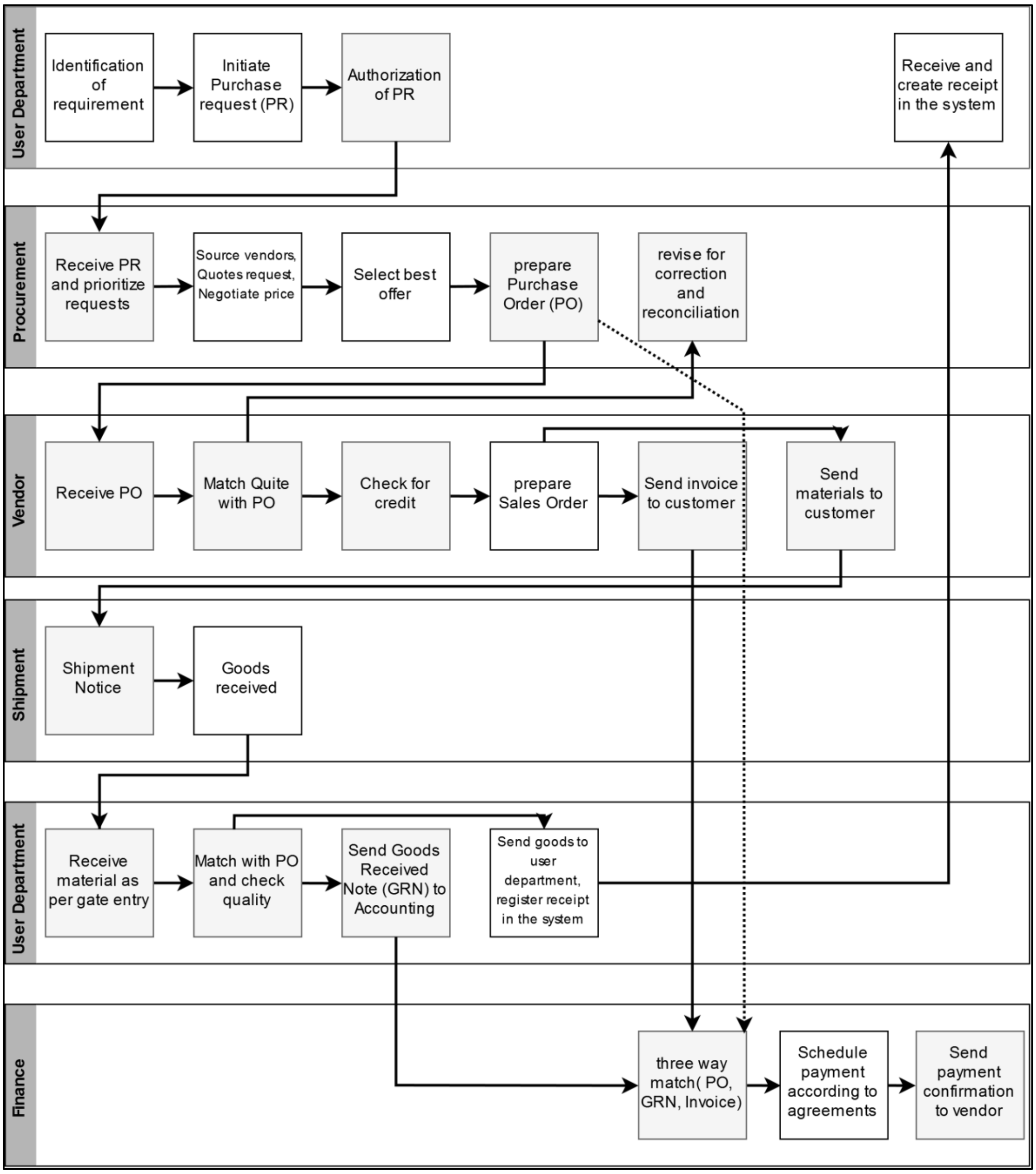

Figure 4-6 Purchase to Pay process

If centralized information systems were used, accounting departments in both organizations will have access to the all information about the purchase request (PR), purchase order (PO), sales order (SO), and goods received note (GRN) and payments would be easily approved since accounting department has all required documents at their disposal. In fact, self-billing services exist already in such systems. A large part of this process is already automated in ERP packages where all operations are applied to a single database. However, in distributed systems, all communications between different parties are done manually. Therefore, matching is required and in some cases, revision for correction and reconciliation is needed. The processes marked in gray are shared amongst different systems, whereas white processes are department specific.

Implementing blockchain in the system will transform the process into the one depicted in (Figure 4-7).

- The process is instantiated by manual identification of requirements or by a smart contract activated by the inventory management system.
- Authorization and final approval of the purchase request requires multi-signature from different participants such as the user instantiating the request, the budget holder, and the department manager in addition to the warehouse confirming that they lack the requested materials. Procurement can be involved in early stages and create a PO. Similarly, the finance department will have a copy of the PO in real time and payment terms can be specified in a form of smart contract.
- If the same order has been done before, it means that its smart contract exists in the contract management component and the purchase order will be issued automatically and validated by the supplier organization by:

- Warehouse: whether they have inventory of the required materials,
- Manufacturing: whether they need to manufacture,
- Sales: for sales operations and records, and
- Finance department for updating their account receivable according to conditions specified in the smart contract.

- The carrier can also validate the shipment transaction. This might be useful in long term relation with carrier

- The carrier together with warehouse, finance, and procurement, validate that the received goods are the same as what is agreed on in the purchase order.
- If this done successfully, payment terms and other legal obligations can be fulfilled using smart contracts similar to self-billing systems.

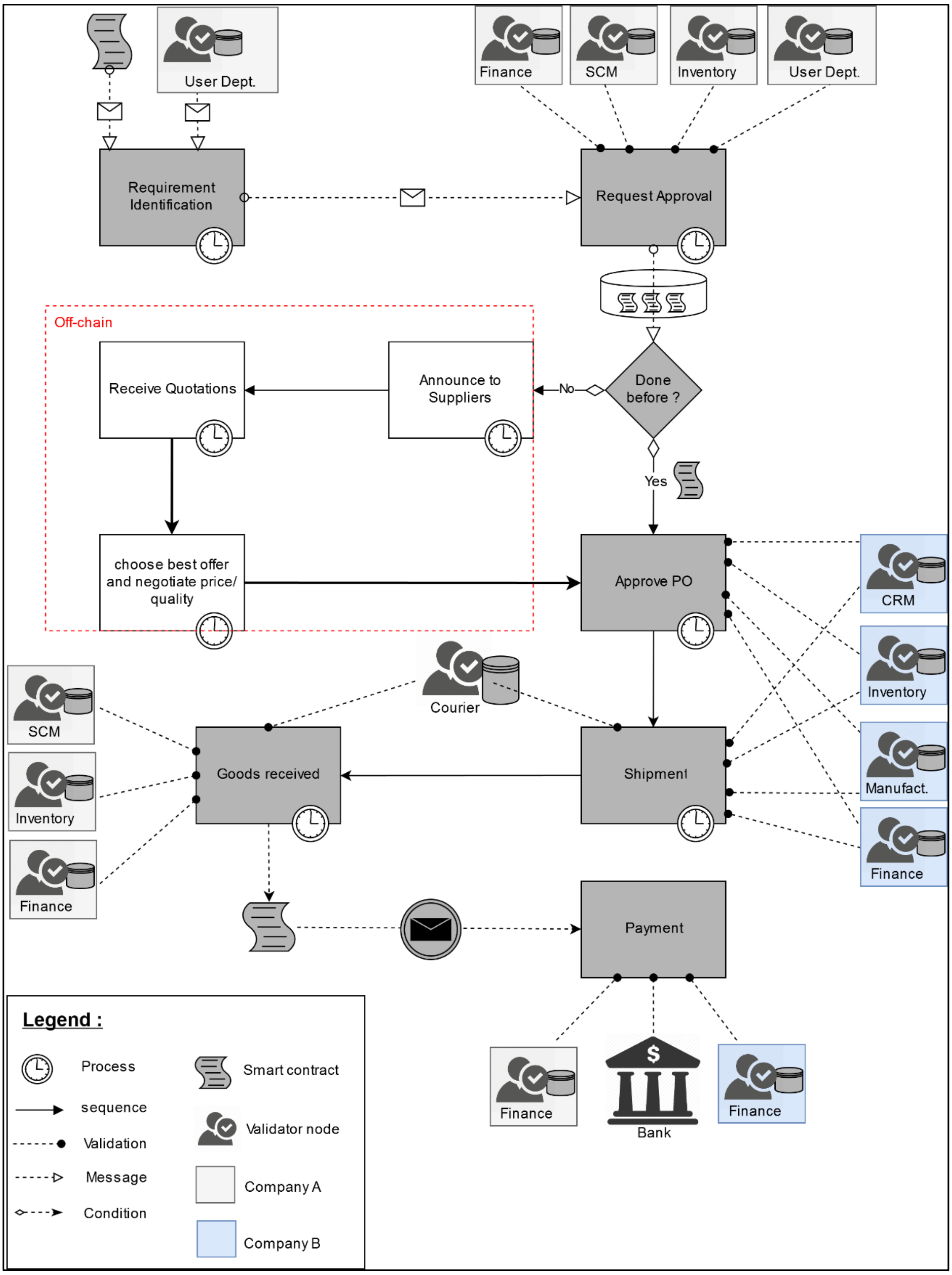

Figure 4-7 Purchase to Pay with blockchain

During the model validation by the procurement consultant (W.D.R.), he suggested that sourcing to suppliers, negotiation, and choosing the best offer are strategic procurement and details about strategic operations should not be revealed to other suppliers. Therefore, these parts of the process must be handled off-chain.

In this way, all different departments involved in the transaction will have the same information about the order even though each node is interested in different facts about the data. In other words, getting everyone to agree on the same data stored in the blockchain will standardize the log data. There will be no need for multiple checking with different department. By adding time stamps, all operations in shared business processes will be easily traceable. In the scenario above, the procurement module closes the loop in purchase process with accounts payable. Similarly, the Customer Relationship Management (CRM) module closes the loop of sales with marketing and accounts receivable. Traditionally, the loops close when invoice matching between purchase order, inventory receipt, and supplier invoice succeeds. If these parties where involved in the process flow and validated all transactions from the beginning, this double-checking will be redundant.

Furthermore, this implementation allows unified data representation and unified communication language between parties. Besides, blockchain is able to validate the consistency of transactions based on rules attached with the transactions in terms of smart contract. If this succeeded, there will be no need for third parties between business partners. For instance, the "invoice provider" in (Figure 4-2 section 4.1) will be eliminated and exchanging business documents will look like one shown in (Figure 4-8).

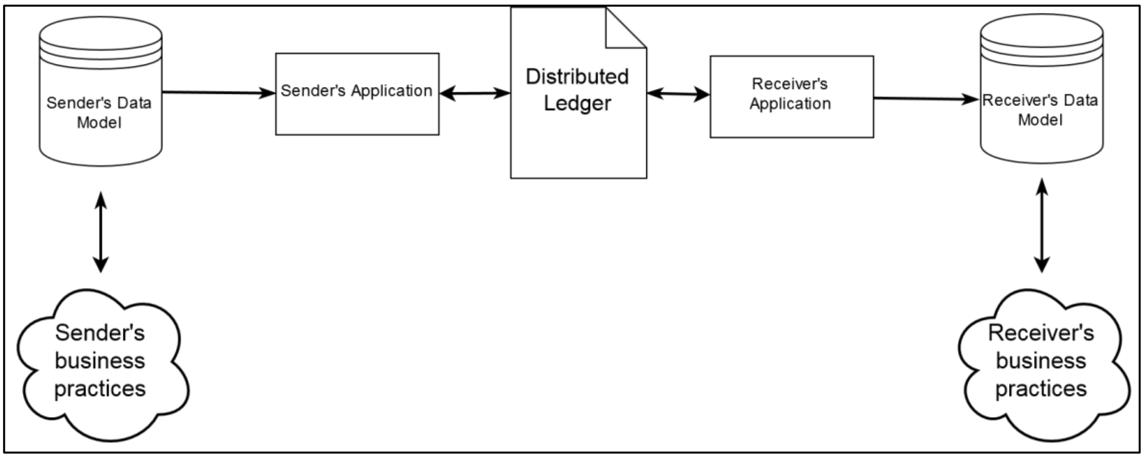

Figure 4-8 Replacing invoice provider with blockchain

In the final iteration, the design concept was introduced to a financial auditor. The purpose of the interview was to validate whether the system will increase trust in the financial auditing process. For this matter, interview questions focused on the current procedures and the measures for trusting the company's records.

During the financial audit process, auditors go through all transactions that have been executed since the last audit. They compare all records and receipts from different systems in order to trace the origin of every transaction. There are regulations for the proper storage and maintenance of organization's records. Although it is possible to manipulate the data from a technical perspective, there are internal and external audit teams who are hired to detect fraud, protect shareholders interests, and provide reliable tax statements. (E. H.).

The financial auditor concluded that the immutable nature of blockchain would "in theory" increase the trustworthiness of the records. However, practically speaking, auditing teams do not even trust other trusted auditor's reports; it is their job not to trust. They also follow standard procedures and legal practices. Therefore, trusting blockchain records requires acknowledging it as trusted system by legislators. Similarly, consultants believe that legal entities need to trust a blockchain vendor in order to trust blockchain technology itself.

To sum up, we introduced in this chapter an architectural design for implementing a blockchain within enterprise systems. Blockchain will be implemented as a pluggable part connected to the whole system though a custom software connector.

Whereas, blockchain did not present significant advantage over centralized systems, preliminary interviews revealed some challenges in distributed systems that can be addressed by this technology. The design was built based on recommendations and subsequently validated by experts who suggested some refinements toward achieving a system that improves data interoperability, efficiency, and tackle some integration challenges caused by organizational silos.

Chapter 5
Discussion and Conclusions

In this final chapter, we will present the discussion about the problem and our findings in addition to the design and its utilities. Afterward we will draw some conclusions about the research in general. Finally, we list the limitations and give directions for future research.

5.1 Discussion

Inter-departmental and B2B communications are crucial for organization to improve knowledge sharing and organizational efficiency. Departments should build and manage relationships based on trust and commitment (Blomqvist & Levy, 2006). Trust and commitment within centralized enterprise system are facilitated by running all enterprise applications on a single database that provides a single source of truth. However, achieving centralization requires organizations to deal with a single ERP provider who offers a complete business solution. These ERP systems are highly configurable and contain series of design trade-off to meet various nuances of the same business cycles / processes. In addition, these solutions mostly provide partners with APIs to access the system. Therefore, an implementation of blockchain to achieve a single source of truth in a centralized enterprise system would be redundant. In addition, these systems provide advanced automation for supply chain, inventory, customer relationship management and many other business processes, which eliminates the need for smart contracts. Moreover, an audit trail can be implemented on specific tables in order to create a log for all operations done on these tables. In terms of security, traditional shared databases use different strategies to improve sustainability and throughput, and reduce latency such as master-slave and multi-master replication. Shortly, there are no advantages of private blockchains over centralized enterprise systems or PBS, as long as all communication resides within one single organization.

In contrast, within decentralized enterprise systems, the IT architecture is completely different. The ICT landscape of an enterprise comprises various systems. On one hand, this diversity is necessary to support specific business operations, but on the other hand, it increases the complexity of integration and interoperability. More complex and changing environments create a greater need for functions to interact and this means working together to build new forms of customer value. Therefore, inadequate intelligence sharing can cause inadequate customer service, trust issues, conflicts, and inefficiencies. According to the literature and expert interviews, a challenge facing organizations is the separation of detailed knowledge of operations. This separation resulted in organizational, functional, and cultural silos. These undermine effective collaboration between different elements in the value chain and subsequently, limiting the response options for emergent problems or crucial insights.

The design proposed in this paper implements blockchain technology as a synchronization mechanism within different organizations or organizational units. This can happen without the need for replacing legacy systems and with minimum business process reengineering. It connects trading partners and service providers to create an ecosystem that is always connected. The data saved on the distributed ledger can span the whole enterprise and iscapable of integrating with the proprietary systems of multiple partners. By enabling a "single truth" in the network, such a design can help companies solve communication problems, facilitate trust, and remove the need for third parties. In scenarios where an entity deals with many contractors, third parties become necessary to handle communication, invoicing, and reconciliation. Smart contracts can handle the legal measures and maintain payment and service level agreement.

Smart contracts facilitate long and standard legal provisions. Through their deterministic attribute, they can provide a great amount of certainty to parties that a contractual condition will be honored by forcing the parties to remain loyal to their respective obligations. Although such an incident is less likely to happen between business partners, having this assurance will support trust between contracted parties. Therefore, smart contracts on a blockchain are used to improve the efficiency and accountability in multi-party business interactions.

Trading partners can capture savings and competitive advantages by fostering networked processes and optimizing the complete enterprise instead of individual functions. In addition, this will drive new ways of thinking and working by enhancing visibility, collaboration, and innovation. Having an immutable ledger will facilitate this and enable participants to track back all transactions and resolve any potential conflict.

In terms of implementation, special attention should be given to permission management. The correctness and privacy preserving purpose of smart contracts must be formally reasoned about. Decisions must be studied carefully when choosing between different DLTs. Different consensus protocols will have an immediate impact on system performance and cost. Choosing between private, public, or consortia affects privacy. The decision to implement multiple chains depends on the type of process i.e. strategic processes should be protected and kept inside the organization. On the other hand, blockchain will provide advantages in communication and knowledge sharing when it is used for operational processes.

In public blockchains, the system introduces economic incentives to encourage participants to be honest. In businesses the incentives for validators and users is maintaining an integrated single source of truth and exchange of knowledge openly between different departments in addition to creating an audit trail and minimize controls.

5.2 Conclusions

Bitcoins is the most successful implementation of blockchain, which has proven to be viable solution to create trust in a trust-less ecosystem without central authority. However, after nine years of its invention, blockchain implementation beyond digital currency are still under investigation. Literature about the topic introduces many promising use cases and anticipates disruption to multiple industries and business models. Nevertheless, there is a lack of empirical studies to support them. Publications by technology enthusiasts created greater ambiguity and cause more confusion to the reader. In addition, the how and why to use blockchain for purposes other than value transfer are often missing. One can notice from examining existing proof of concepts in the market, such as the IBM blockchain that they use digital currency (Ether) and mining protocol. Although they are intended for organization use, they are still about currency transfer. Other solutions offer tokenless blockchain that supports all types of transactions, but there are concerns about their technical capabilities. Overall, businesses are concerned about disintermediating third parties and place their trust in the system.

In this thesis, we took a step further toward understanding the implications of blockchain for enterprise. The design science research resulted in an artifact that implements blockchain technology as part of the overall ICT architecture of EIS to work as a synchronization mechanism between multiple databases within multiple organizations. In order to accelerate the technology adoption, blockchain technology should be inserted into larger systems and it is best to think of blockchains in terms of what will eventually surround them. They will not stand alone, but will function within the core of multiple, increasingly distributed ecosystems. As a result, blockchain would be the "single source of truth," for which organizations are striving. Using blockchain as master database between different applications and different parties will address many integration issues. In addition, using blockchain does not require data a migration project; all relevant transaction data will be stored on the ledger and status will be then derived from it.

In terms of trust, further development and increased adoption will encourage business to trust blockchain. In the web for instance, it has already worked for internet protocols despite the fact that they met resistance by people who felt they were too simple to meet enterprise needs. However, improved connectivity and security measures, have led to mass adoption. Now a day, internet is an essential part of every organization's business. For blockchain, the challenges are privacy and regulatory compliance. However, connectivity is becoming such an imperative of modern business that there is a real incentive to tackle those challenges. Businesses increasingly need to collaborate in real-time and they need to be able to do so using shared, trusted data sources. If the distributed transaction logs provided by blockchain can offer that trusted source then it will have huge value to every kind of enterprise.

5.3 Limitations and Further Research

In this study, we tried to increase the understanding of blockchain technology and provide a solution for enterprises to integrate it within their value chain. However, there were limitations that must be acknowledged. First, the timeframe of the study is limited to six months, which affected the number of iterations. We would perform further refinements to provide more details to address additional integration challenges such as blockchain data model and synchronization with other external systems. Therefore, further research is recommended to perform experimental evaluation for such implementation. Moreover, it is also necessary to investigate further implications for business processes i.e. old processes reengineering or new business processes that might emerge in addition to cost/benefit analysis and constructing technology acceptance models that affect adoption.

Moreover, implementing standards for blockchain data infrastructure and protocols is highly necessary. If we will have different blockchains each has its own data structure, technologies, and protocol, we will end up in a similar situation where there are integration issues between different blockchain platforms.

References

Allison, I. (2016, February). Bitnation and Estonian government start spreading sovereign jurisdiction on the blockchain. Retrieved February 27, 2017, from http://www.ibtimes.co.uk/bitnation-estonian-government-start-spreading-sovereign-jurisdiction-blockchain-1530923

Ametrano, F. M. (2016). Bitcoin, Blockchain, and Distributed Ledger Technology.

Andrews, S., & Manuel, S. (2016, January 16). Blockchain technology: Is 2016 the year of the blockchain? Retrieved February 16, 2017, from https://blogs.thomsonreuters.com/answerson/blockchain-technology/

Back, A. (2002). *Hashcash-a denial of service counter-measure.*

Bahssas, D. M., AlBar, A. M., & Hoque, M. R. (2015). Enterprise Resource Planning (ERP) Systems: Design, Trends and Deployment. *The International Technology Management Review*, *5*(2), 72–81.

Baker, S., & Dobson, S. (2005). Comparing service-oriented and distributed object architectures. *On the Move to Meaningful Internet Systems 2005: CoopIS, DOA, and ODBASE*, 631–645.

Beck, R., & Müller-Bloch, C. (2017). Blockchain as Radical Innovation: A Framework for Engaging with Distributed Ledgers as Incumbent Organization. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Belinky, M., Rennick, E., & Veitch, A. (2015). The Fintech 2.0 paper: Rebooting financial services, in http://www. oliverwyman. com/content: dam/oliverwyman/global/en/2015/jun. *The_Fintech_2_0_Paper_ Final_PV. Pdf*.

Blockchain. (n.d.). Blockchain Markets. Retrieved February 15, 2017, from /

Blomqvist, K., & Levy, J. (2006). Collaboration capability–a focal concept in knowledge creation and collaborative innovation in networks. *International Journal of Management Concepts and Philosophy*, *2*(1), 31–48.

Bollen, R. (2013). The legal status of online currencies: are bitcoins the future?

Buehler, K., Lemerle, M., Chiarella, D., Lal, A., Heidegger, H., & Moon, J. (2015). *Beyond the hype: Blockchains in Capital Markets - McKinsey Working Papers on Corporate & Investment Banking*. McKinsey & Company. Retrieved from http://www.the-blockchain.com/docs/Blockchains%20in%20Capital%20Markets.pdf

Buterin, V. (2014). A next-generation smart contract and decentralized application platform. *White Paper*.

CoinDesk. (2016, June 4). Making Sense of Blockchain Smart Contracts. Retrieved February 7, 2017, from http://www.coindesk.com/making-sense-smart-contracts/

CompuData Inc. (2015, August 24). Understanding the Role of ERP in Supply Chain Management. Retrieved April 20, 2017, from http://www.compudata.com/the-role-of-erp-in-supply-chain-management/

Crosby, M., Pattanayak, P., Verma, S., & Kalyanaraman, V. (2016). Blockchain technology: Beyond bitcoin. *Applied Innovation*, (2), 6–10.

Crypto-Currency Market Capitalizations. (n.d.). Retrieved January 10, 2017, from https://coinmarketcap.com/

Curry, E. (2004). Message-oriented middleware. *Middleware for Communications*, 1–28.

Custers, J. (2015, June). *The Disruptiveness of Blockchain Technology: "AN EXPLORATORY RESEARCH CONDUCTED IN THE DUTCH PAYMENTS INDUSTRY"*. University of Amsterdam, Amsterdam Business School.

Dai, W. (1998). B-Money. Retrieved January 26, 2017, from http://www.weidai.com/bmoney.txt

Deloitte. (2016, December 29). Deloitte Says Consortia Key To Blockchain. Retrieved January 27, 2017, from http://www.pymnts.com/news/b2b-payments/2016/deloitte-blockchain-distributed-ledger-consortia-r3-2017-payment-rails/

Den Haan, J., Albani, A., & Dietz, J. L. G. (2009). An Enterprise Ontology based approach to Model-Driven Engineering. *TU Delft, Delft University of Technology*.

DeRose, C. (2016, January). "Smart Contracts" Are the Future of Blockchain. Retrieved February 14, 2017, from https://www.paymentssource.com/opinion/smart-contracts-are-the-future-of-blockchain

Douceur, J. R. (2002). The sybil attack. In *International Workshop on Peer-to-Peer Systems* (pp. 251–260). Springer.

Drane, J. (2016, October 1). Is blockchain tech enabling next-gen ERP? Retrieved June 2, 2017, from https://www.linkedin.com/pulse/blockchain-tech-enabling-next-gen-erp-jeremy-drane

Duan, J., Faker, P., Fesak, A., & Stuart, T. (2013). Benefits and drawbacks of cloud-based versus traditional ERP systems. *Proceedings of the 2012-13 Course on Advanced Resource Planning*.

Gabison, G. A. (2016). Policy Considerations for the Blockchain Technology Public and Private Applications. Europian Comission. *SMU Science & Technology Law Review*.

Garrehy, P. (2016, April 4). The Adoption of Cloud ERP. Retrieved April 13, 2017, from http://www.rootstock.com/erp-blog/the-adoption-of-cloud-erp/

Gendal Brown, R., Carlyle, J., Grigg, I., & Hearn, M. (2016, August). Corda Platform- Introductory white paper.

Glaser, F. (2017). Pervasive Decentralisation of Digital Infrastructures: A Framework for Blockchain enabled System and Use Case Analysis. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Grau Miró, J. (2016). *Strategic innovation in financial sector: Blockchain and the case of Spanish banks*. KTH Royal Institute of Technology. Retrieved from http://oatd.org/oatd/record?record=oai\:DiVA.org\:kth-189176&q=blockchain

Hasselbring, W. (2000). Information system integration. *Communications of the ACM*, *43*(6), 32–38.

He, W., & Da Xu, L. (2014). Integration of distributed enterprise applications: A survey. *IEEE Transactions on Industrial Informatics*, *10*(1), 35–42.

Herrera-Joancomartí, J. (2015). Research and Challenges on Bitcoin Anonymity. In J. Garcia-Alfaro, J. Herrera-Joancomartí, E. Lupu, J. Posegga, A. Aldini, F. Martinelli, & N. Suri (Eds.), *Data Privacy Management, Autonomous Spontaneous Security, and Security Assurance: 9th International Workshop, DPM 2014, 7th International Workshop, SETOP 2014, and 3rd International Workshop, QASA 2014, Wroclaw, Poland, September 10-11, 2014. Revised Selected Papers* (pp. 3–16). Cham: Springer International Publishing. Retrieved from http://dx.doi.org/10.1007/978-3-319-17016-9_1

Hevner, A., & Chatterjee, S. (2010). *Design science research in information systems*. Springer.

Huls, C. J. (2015). *A scenario planning for interbank payments and Decentralized Ledger Platforms*. University of Twente.

Jentzsch, C. (2017, January 2). Public vs Private chain. Retrieved June 13, 2017, from https://blog.slock.it/public-vs-private-chain-7b7ca45044f

Ken Holman, G. (2017, January 1). What is UBL? Retrieved May 5, 2017, from https://www.xml.com/articles/2017/01/01/what-is-ubl/

Kokalitcheva, K. (2015). Bitcoin: Is digital currency here to stay? Retrieved February 16, 2017, from http://fortune.com/2015/09/21/bitcoin-here-to-stay/

Kondor, D., Posfai, M., Csabai, I., & Vattay, G. (2014). Do The Rish Get Richer? An Empirial Analysis If The Bitcoin Transaction Network. *LoS ONE 9(2): e86197*. https://doi.org/10.1371/journal.pone.0086197

Lamport, L., Shostak, R., & Pease, M. (1982). The Byzantine generals problem. *ACM Transactions on Programming Languages and Systems (TOPLAS)*, *4*(3), 382–401.

Laurent, L. (2017, January). Blockchain Dreamers Will Have a Rude Awakening in 2017. Retrieved February 22, 2017, from https://www.information-management.com/news/blockchain-dreamers-will-have-a-rude-awakening-in-2017

Lewis, A. (2016, February 1). A gentle introduction to smart contracts. Retrieved May 12, 2017, from https://bitsonblocks.net/2016/02/01/a-gentle-introduction-to-smart-contracts/

Lindman, J., Tuunainen, V. K., & Rossi, M. (2017). Opportunities and Risks of Blockchain Technologies–A Research Agenda. In *Proceedings of the 50th Hawaii International Conference on System Sciences*.

Linux Foundation. (2015, December 17). Linux Foundation Unites Industry Leaders to Advance Blockchain Technology | The Linux Foundation. Retrieved January 19, 2017, from https://www.linuxfoundation.org/news-media/announcements/2015/12/linux-foundation-unites-industry-leaders-advance-blockchain

Lotfi, Z., Mukhtar, M., Sahran, S., & Zadeh, A. T. (2013). Information sharing in supply chain management. *Procedia Technology*, *11*, 298–304.

Mehta, N. R., Medvidovic, N., & Phadke, S. (2000). Towards a taxonomy of software connectors. In *Proceedings of the 22nd international conference on Software engineering* (pp. 178–187). ACM.

Menge, F. (2007). Enterprise service bus. In *Free and open source software conference* (Vol. 2, pp. 1–6).

Millar, J., & Brunet, E. (2016). *Bitcoin & Blockchain in 2016 - A Survey of Global Leaders* (Survey).

Morabito, V. (2017). Business Innovation Through Blockchain: The B3 Perspective. In *Business Innovation Through Blockchain: The B^3 Perspective* (pp. 125–142). Cham: Springer International Publishing. Retrieved from http://dx.doi.org/10.1007/978-3-319-48478-5_7

Motiwalla, L. F., & Thompson, J. (2012). *Enterprise systems for management*. Pearson Boston, MA.

MSG.come. (n.d.). Implementation of ERP System - Reengineering vs Customization, Big Bang or Phased Implementation. Retrieved April 13, 2017, from http://www.managementstudyguide.com/implementation-erp-system.htm

Nakamoto, S. (2008a). A Peer-to-Peer Electronic Cash System. Bitcoin.org. Retrieved from https://bitcoin.org/bitcoin.pdf

Nakamoto, S. (2008b, October 31). Bitcoin P2P e-cash paper. Retrieved from http://www.metzdowd.com/pipermail/cryptography/2008-October/014810.html

Ng, K. (2002). *The design of an enterprise information system using Hierarchical Design Pyramid and Web-based Object Oriented Model*. The Hong Kong Polytechnic University.

Nitin Gaur. (2016, June 16). A Blockchain for the Enterprise – A Technical Perspective. Retrieved May 12, 2017, from http://infocastinc.com/insights/a-blockchain-for-the-enterprise-a-technical-perspective/

Nomura Research Institute. (2016, March). Survey on Blockchain Technologies and Related Services.

Oracle Inc. (2013). Bookshelf v8.1/8.2: Summary of Integration Options with Siebel EAI. Retrieved May 23, 2017, from https://docs.oracle.com/cd/E14004_01/books/EAI1/EAI1_UnderstandIntegration7.html

Oracle Inc. (n.d.). Understanding Resource Adapters. Retrieved May 1, 2017, from https://docs.oracle.com/cd/E13222_01/wls/docs92/resadapter/understanding.html

Peng, G. C. A., & Gala, C. (2014). Cloud ERP: a new dilemma to modern organisations? *Journal of Computer Information Systems*, *54*(4), 22–30.

Pilkington, M. (2016). Blockchain technology: principles and applications. *Research Handbook on Digital Transformations, Edited by F. Xavier Olleros and Majlinda Zhegu. Edward Elgar*.

Pinna, A., & Ruttenberg, W. (2016). Distributed Ledger Technologies in Securities Post-Trading Revolution or Evolution?

PYMNTS.com. (2016, December 20). Who's Left In The Dark About Blockchain. Retrieved January 27, 2017, from http://www.pymnts.com/news/b2b-payments/2016/deloitte-blockchain-industry-sector-disruption-financial-services-manufacturing-technology-finserv/

Rik Kirkland. (2016, May). How blockchains could change the world | McKinsey & Company. Retrieved February 21, 2017, from http://www.mckinsey.com/industries/high-tech/our-insights/how-blockchains-could-change-the-world

Ross, J. W., Weill, P., & Robertson, D. C. (2006). Enterprise Architecture As Strategy Harvard Business School Press. *Boston, MA*.

Saunders, M. N. (2011). *Research methods for business students, 5/e*. Pearson Education India.

Shaw, M. (2016, November 24). Blockchain Technology is Real. It's Here Now and Is Already Changing the World. Retrieved February 22, 2017, from https://www.linkedin.com/pulse/blockchain-technology-real-its-here-now-already-changing-michael-shaw

Shrier, D., Larossi, J., Sharma, D., & Pentland, A. (2016). *Blockchain & Transactions, Markets and Marketplaces*.

Shrier, D., Sharma, D., & Pentland, A. (2016, May). MIT_Blockchain_Fin_Services_Whitepaper_Part_One_May_2016.pdf. Retrieved December 18, 2016, from http://cdn.resources.getsmarter.ac/wp-content/uploads/2016/05/MIT_Blockchain_Fin_Services_Whitepaper_Part_One_May_2016.pdf

Simon, H. A. (1996). *The sciences of the artificial*. MIT press.

Swan, M. (2015). *Blockchain: Blueprint for a New Economy*. O'Reilly Media.

Szabo, N. (1997). Formalizing and Securing Relationships on Public Networks. *First Monday*, *2*(9). Retrieved from http://firstmonday.org/ojs/index.php/fm/article/view/548

Tarantilis, C. D., Kiranoudis, C. T., & Theodorakopoulos, N. D. (2008). A Web-based ERP system for business services and supply chain management: Application to real-world process scheduling. *European Journal of Operational Research*, *187*(3), 1310–1326.

Taylor, W., Huls, C., & Mager, C. (2017). Four Blockchain use cases for banks. FinTech.

Trautman, L. J. (2016). Is Disruptive Blockchain Technology the Future of Financial Services?

Tschorsch, F., & Scheuermann, B. (2015). Bitcoin and beyond: A technical survey on decentralized digital currencies. *IEEE Communications Surveys & Tutorials*, *18*(3), 2084–2123.

Von Alan, R. H., March, S. T., Park, J., & Ram, S. (2004). Design science in information systems research. *MIS Quarterly*, *28*(1), 75–105.

Walport, M. (2016). Distributed Ledger Technology: Beyond Blockchain. *UK Government Office for Science*.

Weins, K. (2017, February 15). Cloud Computing Trends: 2017 State of the Cloud Survey. Retrieved June 2, 2017, from http://www.rightscale.com/blog/cloud-industry-insights/cloud-computing-trends-2017-state-cloud-survey

WeUseCoins. (2017). Venture Capital Investments in Bitcoin and Blockchain Companies. Retrieved January 19, 2017, from https://www.weusecoins.com/en/venture-capital-investments-in-bitcoin-and-blockchain-companies/

Wolter, R., & Haselden, K. (2006, November). The What, Why, and How of Master Data Management. Retrieved May 19, 2017, from https://msdn.microsoft.com/en-us/library/bb190163.aspx

Wood, G. (2014). Ethereum: A secure decentralised generalised transaction ledger. *Ethereum Project Yellow Paper*, *151*.

World Economic Forum. (2015). *Deep Shift: Technology Tipping Points and Societal Impact* (No. REF 310815). World Economic Forum.

Wyman, O. (2016). Blockchain in Capital Markets: The Prize and the Journey. *Euroclear, February*.

Young, J. (2017). Deloitte: Banks Need to Form Consortiums For Blockchain to Work. Retrieved January 27, 2017, from https://cointelegraph.com/news/deloitte-banks-need-to-form-consortiums-for-blockchain-to-work