Growth of Blockchain and Smart Contract Technology and Influence on Process Automation

Masaki Minamide
Humboldt Univeristy of Berlin
Email: masakiminamide@gmail.com
March, 2018

I. Abstract

This paper aims to explore the underpining technology behind blockchain and smart contracts, in order to provide evidence on how the adoption of these technologies have grown and continue to do so. It further tries to examine how various applications of smart contracts can and will change business processes and industries to reduce market frictions and the function of middle-men by incorporating a decentralized network which is autonomous and driven by participant consensus.

II. Introduction

In the recent past, a major topic of discussion among the technology enthusiasts has been the rise of cryptocurrencies. While much of the mainstream media interest shown in this area is surrounded by the price hikes of bitcoin and other similar cryptocoins, the fundamental technology concepts driving such applications have taken a back seat. The main aim of this paper is to explore the underpinning science behind the hype, namely blockchain and smart contracts, in order to provide evidence on how the adoption of these technologies have grown and continue to do so. In addition, the paper will also examine multiple application areas where smart contracts are and could be utilized in order to boost transactional efficiencies by reducing market frictions caused by intermediaries.

According to a survey report published by the World Economic Forum, by 2027, 10 percent of the world's gross domestic product (GDP) is predicted to be stored on blockchain technology [1]. This implies that the technology has moved out of its infancy and is attracting more businesses and entreprenuers who not only aim to maximizes the potential of blockchain but also solidify there place in the new market. Subsequently, PwC, in their predictions for blockchain (2016) [2], has highlighted three major trends to be expected;

1) Protection of intellectual property by incumbents as they approach new collaborations with customers, suppliers, and competitors
2) Large financial institutions will start making strategic plans to set parameters for technology risk taking due to new market entrants
3) Market participants will start to develop the processes that surround the transactional layer such as governance, auditing and IT security

III. Methodology

The Methodology used in this paper can be broken down into three distinct steps. **Step 1** introduces the concept of technology adoption and the underlying concepts behind innovations such as bitcoin, blockchain, ethereum and smart contracts. Technology adoption is explained using the three most commonly used models which include; the Gartner Hype cycle, Performace S curve and the Adoption curve. It also helps in giving a perspective on where blockchain technology is on each of these cycles and proving the relevancy of this topic at this point in time. Concepts of blockchain, ethereum and blockchain are based on White and Yellow paper publications on these topics. A white paper is usually a report or publication used to introduce the key concepts and applications of a new technology or issue, while a yellow paper is a more formal academic declaration signifying the research behind a particular topic.

Step 2 involves data scraping of ethereum and smart contract data to give statistics on daily and monthly account creation data and the volume and frequency of transactions taking place on the ethereum blockchain network. This provides an empirical basis to provide evidence of the growing use of the technology and the types of applications being developed on it. Furthermore, specific applications and merits of smart contracts are also highlighted in the section to signify how this innovation is being used to transform existing industries by involving automation processes as well removing intermediaries and transactional frictions which have been embedded in many business processes due to the existence of a centralized network of control and validation.

Finally **Step 3**, gathers the shortcomings and limitations of the wide scale adoption of smart contract technology and attempts to provide possible solutions to overcome such factors which include but are not limited to; security, adoption hesitancy, government regulation and public understanding of the technology.

IV. Technology Adoption

In order to understand the maturity and adoption of new techonologies, the three most popular tools [3] used are :

1) Gartner's Hype Cycle
2) Performance S Curve
3) Adoption Curve

Exploring these tools will help in establishing the concepts driving market penetration and refinement of blockchain and smart contracts (an outcome of this technology).

A. Gartner's Hype Cycle

Gartner's Hype Cycle gives a graphical representation of the expectations surrounding the technology (hype) and the amount of time a particular technology survives in the market (maturity). According to Gartner, each hype cycle can potentially be segregated into five distinct phases: Technology Trigger, Peak of Inflated Expectations, Trough of Disillusionment, Slope of Enlightenment and Plateau of Productivity has highlighted in Fig 1.

The technology trigger phase, as the name suggests, identifies the time when a technological breakthrough took place publicly i.e either via a press release or an academic publication. In the case of blockchain technology, this can be identified as the paper published by Satoshi Nakamoto in 2008, Bitcoin: A Peer-to-Peer Electronic Cash System. Generally this phase of the hype cycle is mostly research based with majorirty of the interest arising from the academic community.

Fig. 1. Garter's Hyper Cycle, Source: Gartner Research (May 2003)

In the recent past, blockchain technology or cryptocurrencies, seemed to have reached the Peak of Inflated Expectations. This period is characterized by the increasing number of market participants trying to take advantage of the marketing hype surrounding the new technology and established companies trying to incorporate it into their existing business models. Evidence of this can be seen by the number of cryptographic assets listed on coinmarket.com, a primary reference website for digital asset developer and market speculators. In a period of 18 months between January 3, 2016 and June 5, 2017, the number of listed cryptocurrencies grew from 551 to 857, an increase of 55.53 percent [4].

When a technology fails to live up to the hype, it falls into the Trough of Disillusionment. This is the phase when less committed investors and market participants exit the sector triggered usually by public failures of some applications of this technology or the diminishing interest of the media. In the case of blockchain/cryptocurrencies (used interchangeably in this section to illustrate the high interconnection between the two), this fall was initiated by the price drop of several cryptocurrencies especially bitcoin caused by multiple factors such as market speculation and uncertainty about government regulations surrounding digital currencies.

Currently, blockchain technology can be understood as entering the slope of enlightenment phase. This is the time period when the technology becomes more widely understood and increasingly refined. Evidence of this can be seen from the fact that bitcoin is starting to gather acceptance as a payment method across various industries ranging from fast food franchises such as KFC Canda and Subway, online booking websites such as expedia.com, cheapair.com and overstock.com while microsoft now allows bitcoin payments for user who want to buy content on Xbox and Windows stores [5] .

On the other hand, blockchain, the underlying enterprise applications technology powering bitcoin and other cryptocurrencies has started to get enterprises interested in the applications of this technology in decreasing process friction. IBM and Maersk have announced a partnership to boost security and shipping efficiency using blockchain technology by introducing a distributed ledger system for all supply chain participants including exporters, cargo handlers (ship, rail, air and road) and regulatory authorities [6]. The intention behind this is to have better risk assessment, verifiable authenticity and ultimately lower administrative costs.

With blockchain rapidly climbing the Slope of Enlightenment, Gartner's Hype cycle predicts that it would soon reach a point of mainstream adoption in the Plateau of Productivity phase. In this time period, the technology is well established and a number of service providers are in the market with expertise to integrate the new mechanics for various businesses. This phase also shows the return of initial investments made by adopters in the form of increased revenue and reduced costs through economy of scale and technology maturity. Gartner does not give a standard time to maturity assessment for new technologies as each new innovation moves at its own speed ranging from four to five years for Fast-track technologies or one to two decade for Long-fuse ones.

B. Performance S Curve

The performance S curve, first introduced by Richard N. Foster in his book Innovation: The Attacker's Advantage (1986), is a function that relates the effort or time spent in the development of a technology with respect to the relative performance or maturity of that technology. The function is similar to the one used in models of innovation diffusion with the name stemming from the shape of the resultant sinusoidal line graph, as illustrated in Fig 2.

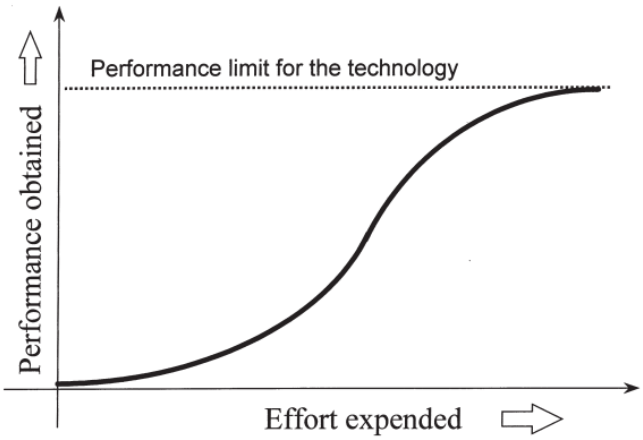

Fig. 2. Performance S Curve, Source: Nieto (1998)

The S curve reveals the evolution process of the performance of technologies and that it systematically repeats itself in all industrial sectors [7]. Therefore, when evaluating a product or an industry, it is crucial to understand where it is on the S-curve in order to identify the possible risks associated for certain phases on it. The S-Curve can ideally be broken down into four stages; Ferment, Take-off, Maturity and Discontinuity.

The Ferment phase highlights the S-Curve pattern of innovation. It is when the product/ industry is completely new and at this stage most of the resources are spent on research and development . The Take-off stage is when the product/industry have been adopted by the early majority and a dominant design has been established. When a technology reaches the maturity phase, the product is heading towards last scale adoption and major players in that market segment are seen to emerge which are spending most of the resources in iteration of the technology, making it more stable and improving the overall quality. The Discontinuity phase marks the performance limit of the technology with an opportunity of an innovation to start a new product life cycle.

In terms of cryptocurrencies, bitcoin has entered the maturity phase with approximately 16.78 million bitocoins circulating in the market in December 2017 with a penetration level of 79.9 percent [8]. Since this cryptocurrency is capped at 21 million, the growth rate is slowing down which is a clear sign of a technology entering this phase.

C. Adoption Curve

While the two previous models gauge the product life cycle through market penetration and the evolution of the underlying technology, the adoption curve, first applied to technologies by Everett Rogers in his book; Diffusion of Innovations (1962), incorporates the social acceptance of innovation in the market place.

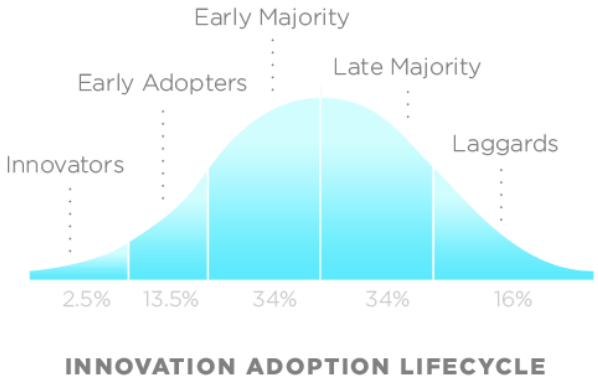

Fig. 3. Innovation Adoption Cycle

The curve in Fig 3 breaksdown acceptance of new technologies in five phases: Innovators, Early Adopters, Early Majority, Late Majority and Laggards.

The Innovators are the first individuals to adopt a new technology. This segment of people are considered risk takers, have ample financial resources and are in constant touch with both scientific innovations and innovators. Early Adopters are the second fastest category of individuals who adopt an innovation. These individuals are considered opinion leaders due to their young age, high social status and a higher level of education then the early majority. This helps them new innovation earlier then most by having a greater sense of future applications of technology in addition to recognizing early mover investment opportunity.

For people falling in the Early Majority category, the time of adoption is significantly longer than the innovators and early adopters. Although these individuals have an above average social status their sphere of influence is limited. Late Majority individuals adopt an innovation after the average member of the society. This stems from the fact that they are financially much more constrained and initial treat technology with scepticism over the useful application of the innovation. The adoption curve reaches it maximum penetration level when the Laggard category of individuals adopt the new technology. This segment is generally older in age and averse to change showing a very limited opinion leadership.

Combing the three types of technology adoption models we get a representation of the position of the new innovations according to the Performance S curve, the Adoption cycle

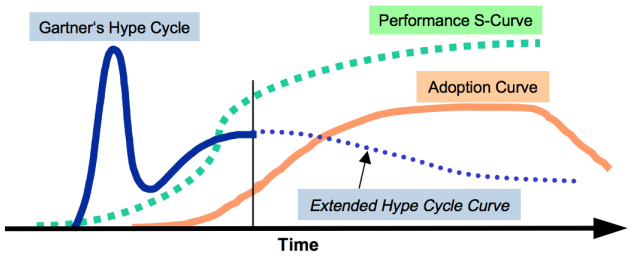

Fig. 4. Overlaying Technology Curves

and Hype cycle. Fig 4. gives a graphical representation of this. As it can be seen by the vertical intersection line, cryptocurrency technology can be seen as entering the Slope of Enlightenment having a greater return on performance obtained on effort expended while moving past the stage of being used by innovators and currently entering the demographic of early adopters, who want to leverage the first mover advantage into better future returns.

V. Blockchain Technology

Having gained perspective on the position of cryptocurrency among the three models of technology adoption in the previous section, it is important to understand the driving technology behind such innovation and establish the fact that blockchain technology has applications in both financial as well as non-financial sectors instead of just being considered the backbone which enables digital currencies.

In essence, Blockchain is a shared ledger technology which allows any participant in the business network to see a system of records (ledger). What this means is that a blockchain is a distributed database of records carrying the list of all executed transactions which have been verified by a majority consensus of the system's participants. The information stored for each transaction is permanent and traceable by any of the system's participants.

This shared or distributed ledger technology offers the intriguing possibility of eliminating the middle man. It does this by filling three important roles; recording transactions, establishing identity and establishing contracts, work which was traditionally carried out by the financial services sector or regulatory bodies. This ensures broader participation, lower cost and increased efficiency thereby reducing if not eliminating market and transactional friction caused by a more traditional monitoring based business network where each party has to maintain there own specific ledger to record all transactions, in addition to hiring auditors to ensure regulation complaint bookkeeping which has proven to be not only expensive but also inefficient and vulnerable to fraud [9].

To understand the blockchain mechanism, it is important to explain the problem which made this solution necessary. As the history of bitcoin and blockchain are closely linked, the problem will be stated in reference to bitcoin (or any cryptocurrency in general). For any transaction to be completed, there needs to be two parties involved, the sender and the receiver each having a public key, an alphanumeric string of 34 characters, and a private key of 64 letters and numbers. Each transaction, protected through a digital signature, is sent to the Äúpublic keyÄù of the receiver, and is digitally signed using the Äúprivate keyÄù of the sender as shown in Fig 4. The entity receiving the digital currency then verifies the digital signature, which implies ownership of the corresponding Äúprivate keyÄù, by using the Äúpublic keyÄù of the sender on the respective transaction [10].

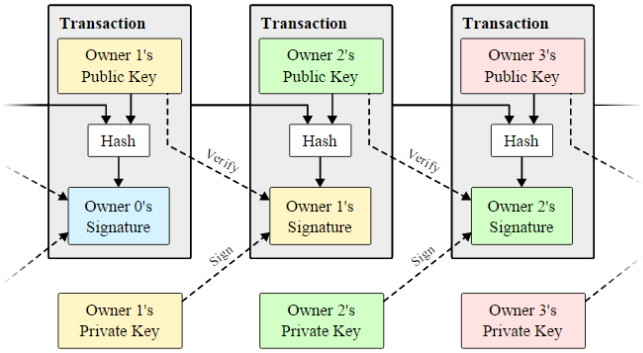

Fig. 5. Bitcoin Transaction Schema

For every single transaction taking place, every node (participant) in the network is informed by recording the transaction in the public distributed ledger after successful verification of the transaction. This verification process can be broken down into two key elements;

1) Ensuring that the sender owns the cryptocurrency
2) Ensuring that the sender has sufficient cryptocurrency funds

The private key used to digitally sign the transaction helps in associating the asset to the participant as this key, an encrypted mathematical derivation of the user's public key, is one that is not known to anyone in the network except the person it belongs to and the system that generates it. The second verification check, to ensure that the participant has sufficient funds is validated through the public key. This public key, which is openly viewable to everyone in the network, is linked to the sender's account transaction history thereby giving a list of all incoming and outgoing transactions associated with the account in the ledger and the therefore the resultant balance at the point of making the new transaction. The schematic view of the transaction is illustrated in Fig 4.

This process of verification and broadcasting the transaction to every participant in the distributed network can cause a possibility of spending the same amount of cryptocurrency twice. This maybe possible since each transaction has to be send to each node in the network via a previous node, there

can be cases where the transaction are not received in the same order as they were generated in. In order to solve this problem, a mechanism is required where all the nodes agree in which order the incoming transactions are sent. Blockchain technology is the solution to this problem in a distributed network.

Blockchain solves the question of double transactions by ordering transactions which happened on the same time and group them together. These groups of transactions are called blocks and when blocks are linked to each other in a linear chronological order the result is a blockchain. The typical internal components of a single block in a chain is illustrated in Fig 6.

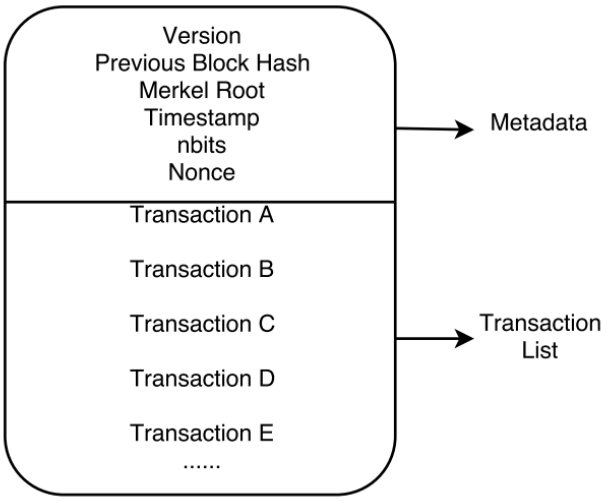

Fig. 6. Single Block Structure

As illustrated, the block can be divided into two main parts; block header and the transaction section. The block header contains the metadata that helps to verify the validity of the block. The parameters contained in the meta data section include:

- **version**: the current version of the block structure.
- **previous block hash**: the reference this block's parent block.
- **merkle root**: a cryptographic hash of all of the transactions included in this block.
- **timestamp**: the time that this block was created.
- **nBits**: the current difficulty that was used to create this block.
- **nonce**: a random value that the creator of a block is allowed to manipulate however they choose.

Due to fact that any node in the network can collect unconfirmed transactions and then distribute it to the network as the next block in the blockchain, with multiple nodes in the network, multiple blocks can be generated at the same time arriving in different orders in different points on the network. In order to circumvent this problem, the process of mining is introduced. Mining is a term which is used as an analogy to the work done in creating a block of transactions. As multiple blocks can be created at the same time, miners (nodes which collect and group multiple transactions in a block) have to solve a certain mathematical problem in order to successfully create a block. Each miner attempts to find a random **nonce** that is part of the block and the results in the generation of a SHA256 hash beginning with a certain number of 0's. The difficulty level of this problem, stated by the **nBits** variable in the metadata, determines the average effort required for solving the issue.

The hash function is a cryptographic technique used to provide a digital imprint of a message's contents, ensuring that the message has not been altered. Hashing is a key element in blockchain technology as it is used to proide integrity of the information contained in a block. When generating a hash, the function analysis all the inputs and produces a unique 256 bit string which will always be the same no matter how often the hash function is run, as long as the inputs remain the same. In case of tempering, the inputs will be altered this affecting the resultant hash function.

In the context of blocks, which contain multiple assets or transaction containing this input information, a merkel tree is used to generate hashes as illustrated in Fig 7.

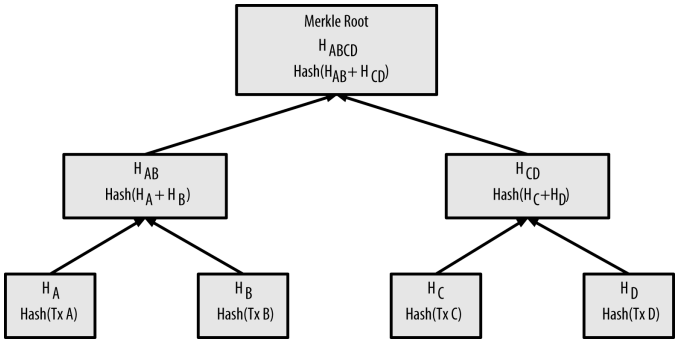

Fig. 7. Merkel Tree Root Hash

Since a block contains multiple transactions $(TxA, TxB, TxC, ..)$, each transaction is run through a hash function giving the resultant hashes: H_A, H_B, H_C, H_D. Now, each pair of hash is run again through a hash function such that H_A and H_B produce H_{AB} and H_C and H_D produce H_{CD}. Finally H_{AB} and H_{CD} are run through the hash function to give the block's merkel root hash i.e: H_{ABCD}. The Merkel root hash, which is also part of the meta data, is unique for each block therefore in case of any tampering in the transactions in the block, the generated hash will change.

The very first block created in a blockchain is referred to as the Genesis block. This block is generally hard-coded for any type of blockchain and does not have the hash of any other block, since it is treated as the initializer of the chain.

Once a block has been successfully **mined**, the consensus of the blockchain is realized and the block is added to the chain. Consensus implies that all nodes in a network verify and validate the authenticity of the transactions and the block against a set of rules which is agreed upon by everyone in the network. Due to the nature of the rules being self enforced, when the blockchain network expands with more participants joining, the overall consensus grows, making it increasingly unlikely for the network to disagree on which block is added to the blockchain. Fig 8 shows a schematic view of a simplified blockchain contaning three blocks of transactions.

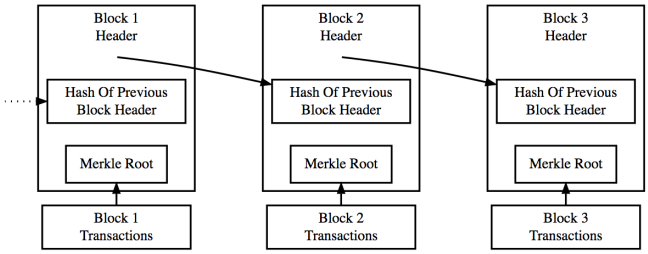

Fig. 8. Simplified Blockchain

As part of the metadata in the block header, the final validation tool for the blockchain is the use of the previous block hash in each subsequent block added to the chain. This act as a critical security feature in case a node decides to change the contents of a block which already is part of the blockchain. Since we know that hashes are generated by a one-way function and that are unique to each set of input, when data is appended in the block e.g:Block 2, the resultant hash of that block will change. Due to the linear nature of the chain, Block 3 has the hash of its parent block, in this case Block 2, thus the change would signify a mismatch in the chain and the network would identify and reject the tampered Block 2. The only way,therefore, to change data in one of the blocks in the blockchain would be to regenerate the hashes for all the following blocks in the blockchain, a task which will take immense amount of computational power [11].

Thus the blockchain mechanism, working as decentralized distributed ledger through a network of computers having an identical copy of the database and changing records by a common agreement based on pure mathematics, establishes four key objectives:

1) Shared Ledger: Append-only distributed system of record shared across business network
2) Privacy: Ensuring appropriate visibility; transactions are secure, authenticated verifiable
3) Validation: All parties agree to network verified transaction
4) Smart Contracts: Business terms embedded in transaction database executed with transactions

VI. Ethereum

Ethereum is an open software platform based on blockchain technology that enables developers to build and deploy decentralized applications. Often Bitcoin and Ethereum are put under the same umbrella although they differ substantially in purpose and capability. Bitcoin offers one particular application of blockchain technology, a peer to peer electronic cash system that enables online payments. While the Bitcoin blockchain is used to track ownership of digital currency (bitcoins), the Ethereum blockchain focuses on running the programming code of any decentralized application.

A. Purpose of Ethereum

The purpose for building a platform such as Ethereum which is designed to be adaptable and flexible is to overcome some of the pitfalls that the Bitcoin blockchain had. These are but not limited to:

1) Lack of Turing-completeness
2) Value-blindness
3) Lack of state
4) Blockchain-blindness

1) Lack of Turing-completeness: A computer is Turing complete if it can solve any problem that a Turing machine can, given an appropriate algorithm and the necessary time and memory. Informally, however, calling a computer Turing complete means that it can execute any algorithm. Although bitcoin scripting language supports a large computational base, it has some limitations, the major missing category being loops. This is done to avoid infinite loops during transaction verification; and can be theoretically overcome by simulating the loop by repeating the code multiple times with an **if** statement. However, this implementation leads to inefficient space consumption by the scripting language.

2) Value-blindness: there is no way for a UTXO [1] script to provide control over the amount that can be withdrawn. For example, one powerful use case of an oracle [2] contract would be a hedging contract, where two participants A and B, put in 1000 worth of Bitcoins and after 30 days the script sends 500 worth of Bitcoins to A and the remaining 1500 to B. This would require an oracle to determine the value of 1 BTC in USD. However, because UTXO are all-or-nothing, the only way to achieve this is through the very inefficient hack of having many UTXO of varying denominations (eg. one UTXO of 2k for every k up to 30) and having O pick which UTXO to send to A and which to B.

[1] A UTXO is an unspent transaction output. In an accepted transaction in a valid blockchain payment system (such as Bitcoin), only unspent outputs can be used as inputs to a transaction.

[2] An oracle, in the context of blockchains and smart contracts, is an agent that finds and verifies real-world occurrences and submits this information to a blockchain to be used by smart contracts.

3) Lack of state: UTXO can either be spent or unspent; there is no opportunity for multi-stage contracts or scripts which keep any other internal state beyond that. This makes it hard to make multi-stage options contracts, decentralized exchange offers or two-stage cryptographic commitment protocols (necessary for secure computational bounties). This also implies that UTXO can only be used to build simple, one-off contracts and not more complex "stateful" contracts such as decentralized organizations, thereby making meta-protocols difficult to implement. Binary state combined with value-blindness also mean that another important application, withdrawal limits, is impossible.

4) Blockchain-blindness: UTXO are blind to blockchain data such as the nonce, the timestamp and previous block hash. This severely limits applications in several other categories, by depriving the scripting language of a potentially valuable source of randomness.

B. Ethereum Virtual Machine

Ethereum is a programmable blockchain. Rather than give users a set of pre-defined operations (e.g. bitcoin transactions), Ethereum allows users to create their own operations of any complexity they wish. In this way, it serves as a platform for many different types of decentralized blockchain applications, including but not limited to cryptocurrencies.

Ethereum in the narrow sense refers to a suite of protocols that define a platform for decentralised applications. At the heart of it is the Ethereum Virtual Machine (ÄúEVMÄù), which can execute code of arbitrary algorithmic complexity. In computer science terms, Ethereum is ÄúTuring completeÄù. Developers can create applications that run on the EVM using friendly programming languages modelled on existing languages like JavaScript and Python. In addition, Ethereum also includes a peer-to-peer network protocol. This means that the Ethereum blockchain database is maintained and updated by many nodes connected to the network. Each and every node of the network runs the EVM and executes the same instructions.

This massive parallelisation of computing across the entire Ethereum network is not done to make computation more efficient. In fact, this process makes computation on Ethereum far slower and more expensive than on a traditional ÄúcomputerÄù. Rather, every Ethereum node runs the EVM in order to maintain consensus across the blockchain. Decentralized consensus gives Ethereum extreme levels of fault tolerance, ensures zero downtime, and makes data stored on the blockchain forever unchangeable and censorship-resistant.

The Ethereum platform itself is featureless or value-agnostic. Similar to programming languages, it is up to entrepreneurs and developers to decide what it should be used for. However, it is clear that certain application types benefit more than others from EthereumÄôs capabilities. Specifically, ethereum is suited for applications that automate direct interaction between peers or facilitate coordinated group action across a network. For instance, applications for coordinating peer-to-peer marketplaces, or the automation of complex financial contracts. Bitcoin allows for individuals to exchange cash without involving any middlemen like financial institutions, banks, or governments. EthereumÄôs impact may be more far-reaching. In theory, financial interactions or exchanges of any complexity could be carried out automatically and reliably using code running on Ethereum. Beyond financial applications, any environments where trust, security, and permanence are important Äì for instance, asset-registries, voting, governance, and the internet of things Äì could be massively impacted by the Ethereum platform.

C. Ethereum Mechanics

The intent of Ethereum is to create an alternative protocol for building decentralized applications, providing a different set of tradeoffs that we believe will be very useful for a large class of decentralized applications, with particular emphasis on situations where rapid development time, security for small and rarely used applications, and the ability of different applications to very efficiently interact, are important. Ethereum does this by building what is essentially the ultimate abstract foundational layer: a blockchain with a built-in Turing-complete programming language, allowing anyone to write smart contracts and decentralized applications where they can create their own arbitrary rules for ownership, transaction formats and state transition functions.

1) Accounts: In Ethereum, the state is made up of objects called "accounts" containing a 20-byte address. The transition of states is triggered by the direct transfers of value and information between accounts. An Ethereum account contains the following four fields:

- Nonce value
- Current ether [3] balance
- Contract code (optional)
- Storage (empty by default)

In Ethererum there are two types of accounts: **externally owned accounts**, controlled by private keys, and **contract accounts**, controlled by their contract code. An externally owned account has no code, and one can send messages from an externally owned account by creating and signing a transaction. In a contract account, every time the contract account receives a message its code activates, allowing it to read and write to internal storage and send other messages or create contracts in turn. Contract accounts work autonomously and have direct control over their ether balance and persistent variables.

[3]Ether is the main internal crypto-fuel of Ethereum utilized in paying transaction fees.

2) Transactions: The term "transaction" is used in Ethereum to refer to the signed data package that stores a message to be sent from an externally owned account. The transactions contain:

- Message Recipient
- Sender' Signature
- Amount of Ether to be transferred
- Data field (optional)
- STARTGAS value
- GASPRICE value

The first three fields are similar to any cryptocurrency including bitcoin, however the ethereum transaction passes some additional information in the data package. The data field, by default has no function but the ethereum virtual machine has an operation code using which enables a contract account to access the data. The contract would read these values from the message data and appropriately place them in storage.

The STARTGAS value represents the maximum number of computational steps the transaction execution is allowed to take while the GASPRICE value determines the fee the sender is supposed to pay per computational step. These values are important to prevent infinite loops or other computational wastage in code made accidentially or for malicious purposes. Each transaction is required to set a limit to how many computational steps of code execution it can use.

Gas is used as a unit of computation with one computational step usually costing around 1 gas although some operations may cost higher amounts of gas due to being computationally expensive, or have larger amounts of data that must be stored as part of the state. The fee system is designed such that a network attacker will have to pay proportionately for every resource that they consume, including computation, bandwidth and storage,therefore, any transaction that leads to the network consuming a greater amount of any of these resources must have a gas fee roughly proportional to the increment.

3) Messages: Contracts accounts have the ability to send messages to other contracts. Messages are virtual objects that are never serialized and exist only in the Ethereum execution environment. A message contains the following:

- Message sender
- Message recipient
- Amount of ether to be transferred
- Data field (optional)
- STARTGAS value

In essense, a message is similar to a transaction, except it is produced by a contract account and not an external account. A message is produced when a contract currently executing code executes the CALL optional code, which produces and executes a message. Like a transaction, a message leads to the recipient account running its code, therby enabling contracts to have relationships with other contracts in exactly the same way that external actors.

4) State Transitions: One of the problems with bitcoin (as mentioned before) was the lack of interediatery states or the ability to do state changes beyond the spent or unspent one. Ethereum has the benefit of a state transition function, where multiple state changes can be acheived, opening up multistep application possibilities. This function can be defined as:

$$APPLY(S, T_X) = S'$$

Where S represents the account state before transition, S' is the account state after that transition and T_X is the transaction enabling the state change with the $APPLY$ function.

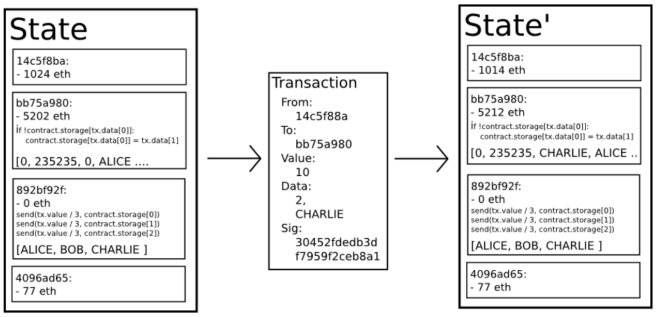

Fig. 9. Ethereum Account State Change

Fig 9., gives a process flow representation of the state transition taking place. During the execution of this function, the following steps are taking place;

1) Check the validity of the transaction T_X. This implies that it has the right number of values including the sender (FROM) and reciever (TO) account, valid signatures, data if present and the amount of Ether to be transferred.
2) Calculate the transaction fee: $STARTGAS * GASPRICE$, subtract the fee from the sender's account balance and increment the sender's nonce. If there is not enough balance to spend, return an error.
3) Transfer the transaction Ether value from the sender's account to the receiving account. In case the receiving account is a contract, the contract's code is run either to completion or till it runs out of GAS.
4) If the transfer of Ether failed due to insufficent balance in the sender's account, or the code execution ran out of GAS, the state changes are reverted, but the transaction fee still goes through to the miner's account.

Messages (created by contract accounts) work similarly to transactions in terms of reverts: if a message execution runs out of gas, then that message's execution, and all other executions triggered by that execution, revert, but parent executions do not need to revert.

5) Ethereum Blockchain: The Ethereum blockchain although similar to the Bitcoin blockchain, stores some additional data in the block. The Ethereum blocks contains a copy of both the transaction list and the most recent state, besides the block number and the mining difficulty of that block of transactions [12]. The basic block validation is in Ethereum is as follows:

1) Check existence and validity of previous block reference
2) Timestamp of newly created block is greater than the referenced block.
3) Check validity of block number, difficulty, transaction root, uncle root and gas limit
4) Check validity of proof-of-work

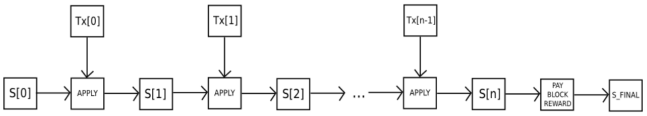

Fig. 10. Ethereum Blockchain

Figure 10, illustrates how the blocks are added to the chain. $S[0]$ is the state at the end of the previous block and T_X is the list of transactions in a block, with n being the number of transactions. For all i in $0...n-1$, set $S[i+1] = APPLY(S[i], TX[i])$. S_{FINAL} is the final block which is a combination of $S[n]$, the final state and the **Pay Block Reward** given to the miner. The final check is made to see if the Merkle tree root of the state S_{FINAL} is equal to the final state root provided in the block header.

D. Decentralized Applications

The Ethereum blockchain architecture gives the ability to build Decentralized Applications or Dapps. Contratry to the widely used centrlized software models, dApps are apps whose server-client models are decentralized. This implies that the computation is performed at each node of a network, with no node instructing another on what to do. Although there is no fixed definition of a Dapp, for an application to be classified as one, it should generally meet the following four criteria:

1) Open Source; Application should be autonomous with all changes being decided by concensus of the users, with no singl body holding a majority vote.
2) Decentralized; All records of the application's operation must be stored on a public and decentralized blockchain
3) Incentivized; Validators of the application blockchain should be incentivized by cryptographic token rewards.
4) Determined Protocol; Application community must agree on a cryptographic algorithm to show proof of value which encapsulates Proof of Work (PoW) or/and Proof of Stake (PoS)

Ethereum can also be used to build Decentralized Autonomous Organizations (DAO). A DAO is fully autonomous, decentralized organization with no single leader. DAOÄôs are run by programming code, on a collection of smart contracts written on the Ethereum blockchain [13]. The code is designed to replace the rules and structure of a traditional organization, eliminating the need for people and centralized control. A DAO is owned by everyone who purchases tokens, but instead of each token equating to equity shares & ownership.

VII. Smart Contracts

In the previous section, the ethereum block chain introduced contract accounts which are not controlled by any external party . These contracts are known as **Smart Contracts** which are essentially agreements fulfilled by computer-generated code explaining the obligations of the interacting parties. Although an official definition is yet to be established, smart contracts can be understood as digital programs, based on the blockchain consensus architecture, which will self-execute when the terms of the agreement are met, and due to their decentralised structure are also self-enforcing and tamper-proof [14]. In many cases, the parties to a smart contract are essentially strangers on the internet bound by this digitally-produced but binding agreement. In effect, smart contracts help exchange money, property, shares, or anything of value in a transparent, conflict-free way while avoiding the services of a middleman.

The transaction rulesets of the smart contract define the conditions; rights and obligations, to which the parties of a protocol or smart contract consent. These rulesets are often predefined and formalized digitally in machine-readable code, with an agreement being reached by simple opt-in actions. These rights and obligations established in the smart contract can now be automatically executed (enforced) by a computer or a network of computers as soon as the parties have come to an agreement and met the conditions of the agreement.

The main advantages of using smart contracts in conjunction with blockchain can be broken down into five major facets;

1) **Autonomy** - agreements can be made by individuals without involvement of a third party such as a broker, lawyer or similar intermediaries to cofirm the validity of the agreement. Thus, this eliminates the possibility of third party manipulation as the execution is managed by the network instead of a an individual or group which may be biased.
2) **Trust** - All smart contract rules are encrypted on a shared ledger therefore there is no possibility of change in rule-sets after a contract has been executed.
3) **Backup** - As smart contracts are stored on the blockchain, the distributed nature allows automatic backup of these contracts on all participant nodes in that particular network.
4) **Savings** - As no intermediary is involved, transactions costs of coordination enforcement are greatly reduced as

no witness is required to validate the agreement between two parties.

5) **Accuracy** - Since smart contract codes are tested and verified through number of edge cases before being accepted in the blockchain, it increases the accuracy of the underlying contract by reducing human error.

A. Smart Contract Trends

To assess the rate of smart contract generation and provide empirical evidence related to the technology adoption cycle of smart contracts and blockchain, contract account data was scrpaed from etherscan.io . Etherscan is a blockexplorer, search, api and analytics platform for the decentralized smart contracts platform; Ethereum. Data was scraped for only those contract accounts which had verified source codes. This prove that there are deterministic and verifiable builds for ther deployed smart contracts. Data which was scraped is from the period: 01-03-2016 to 01-12-2017. The data which was scraped is daily data i.e the no.of smart contracts generated each day. This data is then converted into daily trend chart as well as aggregated on a monthly basis to reflect a more macro level trend. Fig. 11 and 12 illustrate the daily frequency and daily trend of smart contract generation for the period mentioned.

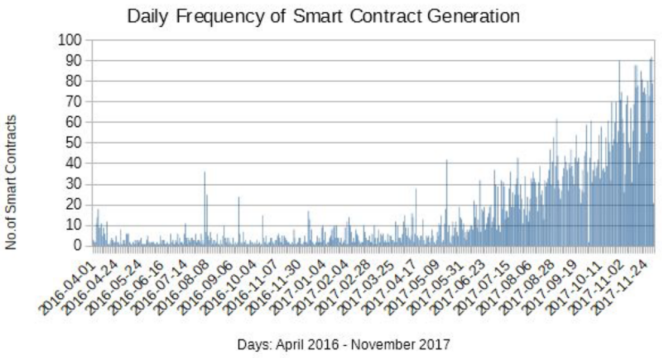

Fig. 11. Daily Smart Contract Generation

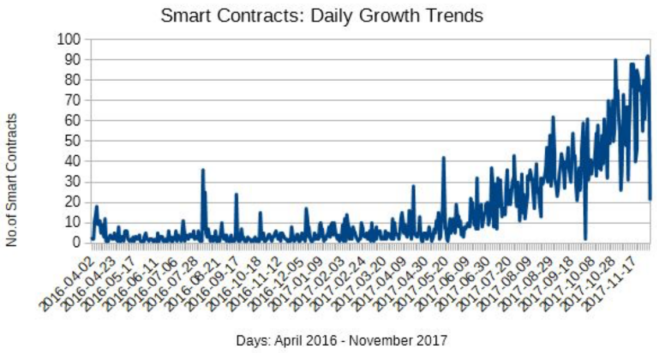

Fig. 12. Daily Growth Trend

From the period 01-03-2016 to 17-04-2017, the daily generation trend is quite stable with contracts being added to the ethereum blockchain, consistently between 5 and 10 except for some periodic spikes near 08-08-2016,06-09-2016 and 17-04-2017 where generation was seen to exceed 20 contracts per day with the highest point reaching just above 40. However after 17-04-2017, Fig 12 shows a major upward trend with positive growth rates carrying on till the end of the analysis period crossing the 90 contracts per day mark on a few times, almost 9 times than the beginning of the period.

Fig. 13 and 14 show the aggregated monthly contract generation data for Etherscan. The monthly data gives a similar representation with a stable generation trend till 04-2017 where contracts being added to the blockchain are well below 500 per month and hovering around 200-300. The only exception seen is 03-2016 when almost 2500 contracts were generated. As this is a lone observation, it can be considered as an outlier caused by some special circumstances. However, after 04-2017 monthly generation start increasing drastically and this trend remains consistent till the end of the analysis period. In Fig 14, the monthly trend chart, 12-2017 shows a big drop but this is unrepresentative of the actual trend as data collected for this month was for only the first few days of December.

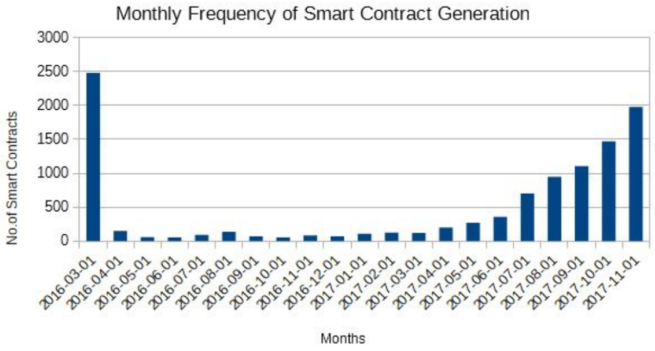

Fig. 13. Monthly Smart Contract Generation

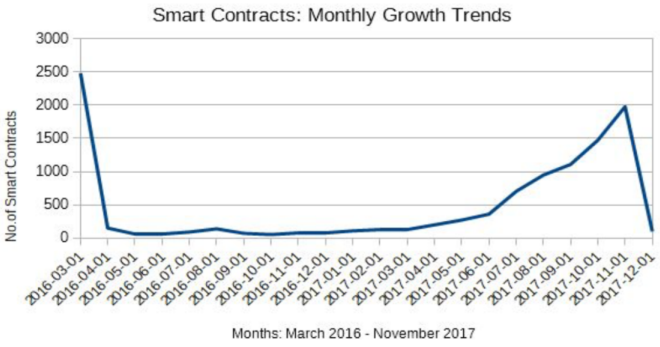

Fig. 14. Monthly Growth Trend

VIII. Merits of Smart Contracts Final

In previous sections, the scraped data from Ether Scan has confirmed the significant growth of smart contracts

deployed on Ethereum blockchain. The increasing number of transactions on the platform of Ethereum blockchain is providing potential services with the platform to deploy their self-executing digital contracts to operate with many merits: transparency, immutability, etc.. These characteristics of smart contract will bring a vital impact in our current economy.

By examining how the major business industry models have been changing last a few decades, how the use of smart contract would change the current industries can be speculated. In the last decade, Business to Customer model such as Amazon and Alibaba dominated the most of the e-commerce business. In this business model, there is only one way of supply and demand flow: from Business-to-Customers, but no otherwise [15]. Then the emerge of Customer-to-Customer business such as Uber and Airbnb started the platform where people can easily find other peoples' demand and supply by using the platform. People can share their flats and buy and sell depending on other users interests, the business model of which is called Sharing Economy.

As of today, both of the Business-to-Customer and Customer-to-Customer business require the existence of centralized platform to bridge supply and demand by mediating the interest of buyers and sellers. This decentralized platform which is fully controlled by TTP("Trusted Third Party") leads customers to being exposed to inevitable disadvantages in terms of security, privacy and costs. In order to use those sharing companies' platforms, users must separately register their account on each platform they want to use. To do so, users must give away their private information to the TTP such as name, age, address, credit card information and so on. The data of customers interests and transactions are also stored in the companies' database server which can possibly be attacked by hackers or sold to other companies.

Furthermore, users are usually charged a variety of fees such as transaction fees and account fees to use the platform. It is also considered that many companies intentionally either make the precise transaction/commission fees difficult to notice or do not make it open for public at all. However, in the case of Uber, it is estimated approximately 20% to 30% [16] of commission is taken from drivers' profit. Thus, there is always risks for users to use the current model of centralized platforms.

One of the most innovative things that can be achieved with the use of Smart Contract and Blockchain-based technology is that customers no longer need to rely on Trusted Third Party to operate platforms. As Smart Contract is often referred as self-enforcing contract due to the fact that all the transactions are autonomously executed without interference of third party, it enables to build a Peer-to-Peer and fully decentralized architecture [17], which has potential to eventually change the incumbent economy model to the one which everyone equally has an access to their own personal data which would never be able to be altered by any third party, by eliminating the need for TTP, and there is no need to pay fees to TTP.

What can be achieved with integration of Smart Contract and BC is number less. Thanks for the special characteristics of which such as cryptography, digital signature, PoW, P2P, etc.., it makes it possible to achieve these things for example. [18]

- Application can be excuted by script
- Double payment is prevented by continuity of chained node
- Data is tracable, highly transparent and immutable
- Zero server down time and low maintanance cost
- The ecosystem is kept from malicious users without superviser

The use of smart contract enables almost every field of economy to receive merits from integrating their incumbent business model with smart contract. Businesses based on DL or Smart Contract can become more efficient, cheaper, easier and faster by dramatically reducing TCs(Transaction Costs) which can broadly be categorized into these four factors. [19]

- Physical presence: physical-location offices, employees to implement work routines
- Organizational Structure: the management function which is needed to supervise employees, and business-hours limitation, office-culture based decision making and processing practices
- Business IT systems:old legacy information system
- Payment and settlement systems: legacy payment and remittance system

These Transaction Costs of hallmark characteristics in the incumbent economy can drastically be lowered with integration of DL and Smart Contracts in the businesses [19]. The incumbent businesses potentially receive a great deal of advantages from shifting their traditional business model to new emerging business model, but at the same time, we also need to anticipate new emerging risks and limitations which might hinder the technology from being assimilated into the economy at the rapidest pace. It is important to know that while these special characteristics that DL based technology and Smart Contracts possibly change businesses in nearly every fields of industry, in some fields more new risks are enhanced and advantages are lowered. Both decentralized business models and decentralized business models have advantages and disadvantages. Thus, entrepreneurs, customers, regulators should know what it will brings to the economy and change it into. In following sections of this paper, in order to deepen the understanding of how it actually be used in economy, practical applications of smart contracts in various business industries and already existing use cases will be introduced. Then, limitations currently hampering the

diffusion of smart contract would follow.

IX. Applications

A. Application Fields

According to the research conducted by Ministry of Economy, Trade and Industry of Japan in 2016 [18], Blockchain technology will potentially bring a social innovation in very broad area. In the research, Smart Contract and BC technology are said to offer these following social impacts and fields:

- Value circulation: value is visualized and tradable on decentralized trading platform, then the data is safely stored on Blockchain (e.g. regional currency, digital voucher, point service)

  Visualized value/points will be traded outside of the ecosystem in which the points are issued. In the result, the points circulation in a number of different ecosystems would make the points to take a role of digital currency,so that they become more valuable than when they had been issued.

  POS(Point of Service) acquiring the function of deposit and lending will enable not only banks but also private corporations to create credits by issuing points as collateral.

- Decentralization of licence, registration, certificate: physical situation of lands, ownership, rights are registered, managed, and proved on Blockchain (e.g. land registration, medical records)

  Land registry, patent, system managed by goverment will be replaced with distributed system, which leads to reduction of work loads of local govermental administry and government. The process and verification documents needed for identity authentification will be easily managed.

- Highly efficient sharing economy: idle assets, transference, supplyers, users infomation of assets are stored on Blockchain (e.g. digital contents, ticket services, C2C auction)

  Besides maximization of idle assets' occupancy rates is achieved, improvement of efficiency in management of usage authority such as renting cars, flats and bicycles is expected. Ultimately, the environment where C2C transactions are completed without the intervention of the current owner of sharing platform will be constructed. As the border between producers and consumers disappears, the idea of "prosumer" –A consumer who becomes involved with designing or customizing products for their own needs –becomes generalized.

- Transpaent, highly trustable, efficient supply chain: from what the product is made of, how it is manufactured, supplyed, sold are tracable on Blockchain (e.g. retailing, management of precious metal goods, authentification of art works)

  Unbundling circulation of goods will be achieved by the enourmous amount of data of supply chain and supplies in storage being shared through DLs. As electronic goods gets connected to Internet due to the rapid growth of IoTs, corporations eventually will be able to track product lifecycle after the goods have been sold to users, then it will be easy to continue after-sales service(e.g. software update, automatic reordering).

- Fully automated process, trade: contract infomation, fulfillment contents, prominent process are stored on Blockchain (e.g. testament ,IoT, power service)

  The majority of back-office works(e.g. contracts, transaction, payment settlement, decision makings) will be replaced. The combination of IoTs and Smart Contract makes it possible to construct the system where costs of public services is more accurately reflecting beneficiary's burden. (e.g. Tax collection according to the amount of garbage and road usage)

X. Case Study

A. Payment/Settlement

Payment and settlement are the most fundamental ways of using Smart contract. As explained previous sections, we no longer need to rely on TTP to complete a P2P transaction because of the fact that Smart contract is capable of autonomously receiving cryptocurrency and sending to the designated address.

Sending and receiving cryptocurrency are the most fundamental functions which Bitcoin transaction has. As long as user's intention is to either send or receive money, Bitcoin has enough functions. However, when it comes to remotely purchasing or renting goods, we need to confront a trust problem. Bob wants to purchase a TV from Alice whom he cannot directly meet to trade. If Bob sent Bitcoin to Alice before Bob make sure TV is sent, Alice might betray and ends up stealing money. With these limited Bitcoin functions, we have no choice but either use centralized platform which mediate the transaction or risk a betrayal.

Unlike Bitcoin blockchain, Ethereum blockchain is capable of executing code of arbitrary algorithm complexity. Owing to EVM(Ethereum Virtual Machine), it makes it possible to let transactions to have simple to complex payment functions, such as purchase goods with deposit function and rent goods

with time limit.

1) Purchase Contract: In the following script, basic functions and flow of purchase contract are written. First, in order to encourage a seller to not betray a buyer, the seller has to send Ether in the constructor function of the contract. As shown on lines 7-8, the value provided must be divided by 2, otherwise the contract throws an error. Thus the actual price of the item is only half the value provided in the constructor. Buyer also has to transit a designated amount of deposit together with the actual item price when the buyer executes the confirmPurchase function on line 23.

After buyer confirmed purchase, the state of the contract is then set to "Locked" on line 25. Namely, it means the funds of the 2 parties are locked in the contract, and it is unlocked only after the buyer executes itemReceived function on line 28. Locking of deposits ensures that no monetary advantage is in the hand of any party in order to prevent betraying other side, since both have deposited the same amount of money in the contract, and none of them can get their deposit back as long as the other party does not follow the contract.

The seller is able to abort the purchase contract as long as the contract is in the state Created on line 14, which means that buyer has not confirmed the purchase. Abort function inherit onlySeller function. As soon as the seller execute abort, the balance stored in the contract is instantly refunded to the seller. The state of the contract is set to Inactive when the buyer executes confirmReceived function, then any further interaction with the contract is possible. The deposit is refunded to the buyer and the remaining balance in the contract is sent back to the seller.

In addition, the state is changed to inactive before the transmission of Ether in the send function on line 30-32. This code order is important because of the fact that otherwise an attacker can call the function again from its fallback function. Every change of the state of the contract is stored in event log in order to let the client app to alter its user interface which informs the user: the buyer or the seller that the change of state has been made.

```
1  contract Purchase is TradeContract
2  {
3    function Purchase(string _title,
4      string _description, bool _verify,
5      bytes32[] _imageSignatures) payable
6      TradeContract(_title, _description,
7      _verify,msg.value / 2, msg.value
8      / 2, _imageSignatures){
9      if(2 * price != msg.value)throw;}
10     event purchaseConfirmed();
11   event itemReceived();
12     function abort()
13   onlySeller
14     inState(State.Created){
15     aborted();
16     state = State.Inactive;
17     if(!seller.send(this.balance))
18     throw;}
19     function confirmPurchase()
20     inState(State.Created)
21     require(msg.value == 2*price)
22     payable{
23     purchaseConfirmed();
24     buyer = msg.sender;
25     state = State.Locked;}
26     function confirmReceived()
27     onlyBuyer22inState(State.Locked){
28     itemReceived();
29     state = State.Inactive;
30     if (!buyer.send(deposit)||
31     !seller.send(this.balance))
32     throw;}
33 }
```

2) Rental Contract: Secondly, Rental Contract which is used in sharing economy between Customer and Customer.

In the renting contract, buyer assumes the role of a renter and the seller assumed as the role of a hirer. The hirer defines details of the rent item together with its renting price and deposit in constructor of the contract. Renting price is provided in wei per second. As long as the contract is in the Created state, hirer can execute the abort function.

```
1  contract Renting is TradeContract{
2    uint256 private rentingFee;
3    uint256 private start;
4    function Renting(sting _title,
5    string _description, bool _verify,
6    _deposit, _rentingPrice,
7    _imageSignatures){}
8    event itemRented();
9    event itemReturned();
10   event paymentRequested();
11   function abort()
12   onlySeller
13   inState(State.Created){
14   aborted();
15   state = State.Inactive;}
16   require(msg.value == deposit)
17 payable{
18   start = now;
19   itemRented();
20   buyer = msg.sender;
21   state = State.Locked;}
22   function calculateRentingFee()
23     returns(uint256){
24     if(state == State.Inactive)
```

```
25        return rentingFee;
26        if(start == 0)
27        return 0;
28        uint256 rentingTime = (now-start);
29        return price * rentingTime;}
30        function returnItem()
31        inState(State.AwaitPayment)
32        onlyBuyer
33        require(msg.value >= rentingFee)
34      payable{
35        itemReturned();
36        state = State.Inactive;
37        uint change = rentingFee-msg.value;
38        if(!buyer.send(deposit + change) ||
39        seller.send(this.balance)) throw; }
40        function again
41        function reclaimItem()
42        onlySeller
43        payable
44        require(state == State.Locked ||
45        state == State.AwaitPayment){
46        rentingFee = calculateRentingFee();
47        paymentRequested();
48      state = State.AwaitPayement; }
49    }
```

In the rental contract, the buyer and the seller are respectively replaced with the renter and the hirer. Firstly, the hirer defines details of the item for renting with its renting price and deposit in constructor of the contract. Renting price is provided in wei per second. Similar to purchase contract, hirer is capable of executing the abort function as long as the contract is in the Created state.

When the renter executes the rentItem function on line 14, the contract checks if the renter has sent enough Ether for the designated amount of deposit. If so, state of the contract changes to "Locked" and renting time is initialized according to the time. When the renter wants to return an item, the hirer executes the reclaimItem function on line 41, then it calculates the renting fee according to the renting time and the price per and sets the state of the contract to "AwaitPayment". In the "AwaitPayment" state, renter can execute returnItem function on line 30, which then returns the deposit and the change to the renter. The calculated renting price is sent to the hirer. In the AwaitPayment state, the hirer can also execute the reclaimItem function again in the case the renter does not pay in time.

B. IoTS

1) The ever-expanding growth of IoTs: Ubiquitous internet connectivity and the ever-expanding capacity of cloud computational capacity have largely contributed into the exponential growth of internet-connected devices. As of today, approximately 23B devices worldwide are installed with base of IoT, and the overall IoT market is annually worth about one billion US dollar at a point of 2017 [20]. The number of IoT device is forecasted to surpass 30 B in 2020, up to 100 B in 2050. [21]

How the expansion of IoT usage in daily life will change the world we currently live in is limitless. Our day-to-day objects are installed with sensor tags which are machine-readable and transmit raw data of physical world to the Internet [22]. Those objects with sensor tags will liquify physical world, which leads to the physical world potentially becoming as liquid, personalized and efficient as the digital world [20]. In the current economy, there are an excessive amount of physical assets that are being idle in warehouse. As the transformation of the physical world into digital one proceeds, IBM claims that those idle physical assets will become available for instant usage, search, and payment. In the result, new economy and business models will be born, where physical assets are connected with each other, and an easy access to a vast amount of raw datas becomes available to those who are authenticated.

On the other hand of the unstoppable growth of IoTs, there have been already problems which hinder corporations and users from willingly adopting the technology in the incumbent industry. According to the study conducted by Bain and Company, Inc., the major two reasons why IoT buyers are hesitant of using IoT solutions are security concerns and high price.

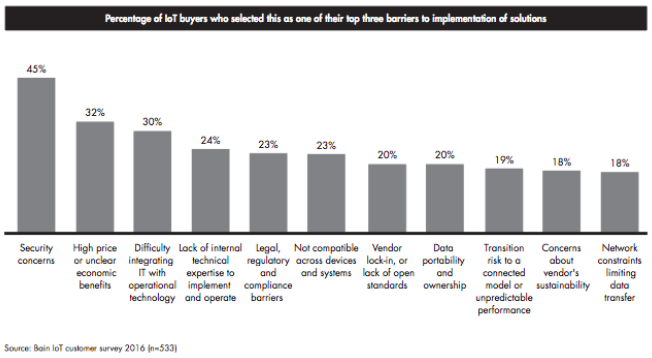

Fig. 15. Barriers of implementation of IoTs, Source: Bain IoT customer survay 2016

2) Two big barriers: Security and High Cost:

- Security Concern The largest problem when it comes to integrating IoTs into incumbents is security concern, with 45% of the whole survey subjects answering in the study [23]. Internet was originally created based on trust. However most solutions and services as today give users no choice but provide the centralized authorities, service providers, manufacturers, governments with the ability to access user data on devices without any notice.

Current security models based on the reliance on the secrecy of the design or implementations- often referred to "security through obscurity"" [20], is obsolete and must be replaced with substitutional solution. But at the same time, open sources will lead to a threshold to vulnerabilities to exploitable weaknesses, being attacked by malicious users. As AI speakers, automated automobile, home locks will become very vital parts of our daily life, there are chances they become potential targets of hackers.

- High cost Even though possibilities of IoTs might look very attractive for most industries, the costs and return of investment are a prominent concern [23]. Due to service costs for middleman of transaction and high maintenance fee of infrastructure to keep centralized clouds and server farms running, costs are extremely high. To add to this, the historical mainframe of business model in which suppliers decide pricing considering product life cycles is also allowing industries to stick to the old business model. Historically, buyers are expected to buy a product after the product cycle ends due to the fact that the out-dated software and hardware may become a target of hacking. With IoTs' merits: after-sale support including updating software will results in corporations' increased burden to cover longer term's support and reduction of sales profit [20].
  In order to foster the expansion of IoTs in incumbents, user privacy and anonymity must be kept secure, and any effective means of cost reduction must be taken.

3) Blockchain as a solution: In the IBM executive report [20], author strongly points out the importance of shift of IoT's current architecture towards a decentralized one, in order to tackle security problem and high costs. For a potential technology strategy, the combination of blockchain technology and IoT is mentioned due to blockchain's decentralization and cryptography characteristics. As explained in before, distributed ledgers of blockchain hold a history of every transaction made, and it is always accessible to all the participants without no downtime. Cryptography used for authentication purpose of transactions keeps records nearly impossible to be altered by a malicious user. Blockchain solution enables secure software, firmware updates, and direct file sharing between peer to peer devices by eliminating need for trust. In the result, the transition from "security through obscurity" to "security through transparency" is achieved [20].

Blockchain will be also a solution to solve the cost problem by sustaining successful decentralized architecture between peer to peer devices. By securely accessing idle devices' computing power through blockchain's consensus verification, it is possible to harness the terabytes of storage and bandwidth for various uses.[IBM] This will drastically reduce costs of installation, maintaining centralized data server, and service fee for middlemen, resulting in lowering the barrier associated with monetization.

4) Slick.it: next generation sharing economy: Slock.it provides solutions with the combination of blockchain and IoTs in the next era of sharing economy. What they are currently developing is Ethereum blockchain based sharing platform called USN(Universal Sharing Network). On this platform, prosumers and businesses can securely and fairly rent, sell or share any objects with blockchain address through blockchain. Moreover, any existing smart object can be connected to the USN without any hardware and software modification, which lowers the entry barrier for already-existing businesses and individuals' assets to move to this sharing economy.

A successful launch of USN is likely to disrupt incumbents of the sharing economy by making trustees sharing platform available for everyone; enabling secure P2P/D2D payment; removing all forms of middlemen. USN aims for making completely autonomous sharing economy where objects are not only being traded but also one object can pay Ether to other objects, which is a crucial factor to achieve a autonomous decentralized IoT architecture.

COO of Stock.it claims that self-driving vehicles and other half-automated objects are not truly autonomous yet [24]. Self-driving vehicles can drive themselves from start point to destination point, however it requires centralized supervising for recharging/refueling, payment and maintenance. As mentioned previously, centralized supervising costs are very expensive and this is one of the biggest reasons why many companies are reluctant of joining sharing economy. On the other hand, thanks to Slock's software operating on Ethereum blockchain, any program can be also deployed as a form of Smart contract. In order for autonomous operation, a vehicle queries through designated contract ABIs to find a function named 'buyElectricity()' for example, which has two parameters: 'cost_per_kw', 'location'. In the autonomous workflow without human supervision achieved by Smart contract, all operations including fueling/recharging, payment, maintenance are operated at the maximum efficiency around the clock.

C. Games

1) The problems in Video Game industries: Video game industry is one of the biggest industry market in the entire entertainment market. The global video games market is estimated to be worth 108.9B, being larger market than music and cinema industries combined [25]. Besides its already grown market, its annual growth is still undiminished, with approximately 6.2% on average during 2016-2020 [25]. As the number of available video games is becoming enormous: there are around 1M games on Google Play platform, $800,000$ on Apple Store [26], and 17,000 on Steam, the business

model that buyers download from those gaming platform is becoming mainstream.

However, due to the fact that as of today the lowered entry barrier that anyone can make video games and publish them on platforms has created new problem in the market. The problem is that now the competition is too strong that makers must make more efforts into marketing of the games besides making them good. Due to the ginormous number of games available on platforms, it is not rare that video games with high quality being not found and paid as much attention as it should have been.

If the current situation continues, video game industry is going to be monopolized by large video companies which have the power and resources to advertise their video games effectively, which would results in an elimination of individual game producers. To add to this, even big video game companies have to suffer from the elevated cost for promotion. Money spent on advertising video games can be up to $20B globally which is about 18% of the whole game industry market cap [26]. In the result, ads platform such as google and apple takes the budget spent on promotion as a middleman.

In addition, to keep the cashflow going for certain period of time is very crucial for any corporation, especially small ones. Almost all those distribution platforms(App Store, Google Play, Steam) have delay of payments after the sale was either successful or unsuccessful, which can be up to 60 days.(GameNation) The sales profit also needs to meet minimum amount to be withdrawn. If not, it is shifted to next cycle of payment(payment is usually done on certain day in month). It is important to know that besides game studios and individual developers are suffering from these current problems, up to 30% of their sales profit is taken by the distribution platforms [26].

2) How GameNations solves this problem: GameNation is a automated platform where gamers, developers and influencers are connected in a way that no middleman takes a big proportion of profit any more, and the new cashflow system programmed in their Smart Contract makes very efficient cashflow among gamers, developers and gamers possible. Influencer is a very critical role in economic cycle in this case: producer is game developer, endorser is influencer and buyer is gamer. In recent years, many games are advertised by influencers who plays the video game and upload or live stream on video platforms such as Twitch and YouTube. Usually, depending on how many downloads and purchases were made through their promotion will decide their dividends to receive.

In GameNation's distribution and promotion model, a game is listed on their distribution platform deployed on ethereum Blockchain by a game developer, with designated influencer agreeing on cooperation. Each influencer has a freedom to choose individual agreement and smart contract that determine the content of their future revenue conditions. Influencers who actively engage in promotion of the video games by creating video content(reviews, gameplay, montages) and posting or live streaming it on their channel are likely to be rewarded for high conversion rates.

As it can be seen in fig.16, a gamer can buy game by sending ether and in return receives game key for Steam or GOG. Smart Contract automatically executes funds distribution that goes directly and almost instantly to those two parties: game developer and engaged influencer(on their platform, 89-90% goes to game developer, 5-15% goes to influencers). Buyers also have freedom to choose other smart contracts: one where more proportion is distributed to game developer or one where a split goes to another influencer). This approach provides with a new way of discovery means through giving incentives to influencers. There is no need for middlemen since the game is bought almost directly from the game developer due to Smart contract's autonomy.

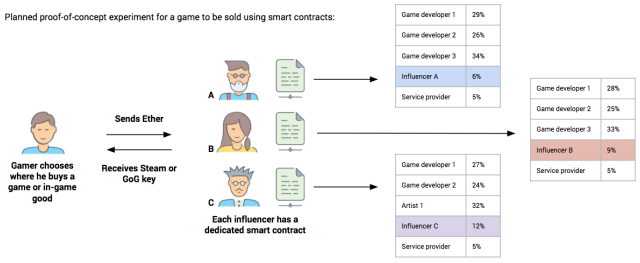

Fig. 16. GameNation's Distribution Platform Model, Source: GameNation

The transparent and fair way of fund distribution claimed on unalterable Smart contract gives much higher incentives to both influencers and game developers. In the result, influencers are much mode deeply involved in game promotion, game developers keep their cash flow stable thanks for instant money distribution, and gamers have option to support their favorite influencers by choosing Smart contract.

Influencers who actively engage in promotion of the video games by creating video content(reviews, gameplay, montages) and posting or live streaming it on their channel are likely to be rewarded for high conversion rates. The transparent and fair way of fund distribution claimed on unalterable Smart contract gives much higher incentives to both influencers and game developers.

3) Games coming next generation: what is CryptoKitties?: CryptoKitties is one of the very first successful dApps(decentralized app) to be build on Ethereum blockchain. After its release to the public on 28th Nov. 2017, over 50,000 digital cats and $6.6M of transactions were traded. The

craze led to slowdown problem in Ethereum blockchain as the transaction on CryptoKitties' smart contract amounted up to 15% of new computations on the Ethereum network [27].

Cryptokitties is a blockchain game in which players collect, breed, and trade Cryptokitties, which have a very unique digital genome that determines what attributes the cat have. In order to create unique cats, the technology called GA(Genetic Algorithm) [28] is used. GA, as it can be inferred from the name, works in a similar way as real biological genetics work like human's DNA. As DNA store all the biological information of humans such as eye color, hair color, and skin color, GA-binary number- stores attributions of cats such as eye color, fur color, curly or straight hair, facial expressions, and background color. While the average sale price of trading CryptoKitties is about $70, Kitties made of rare combination of these attributions are traded at high price-the highest price traded to the date is $110707(253.3368 ether, current value in usd is $181135) on 7/Dec.2017 [29].

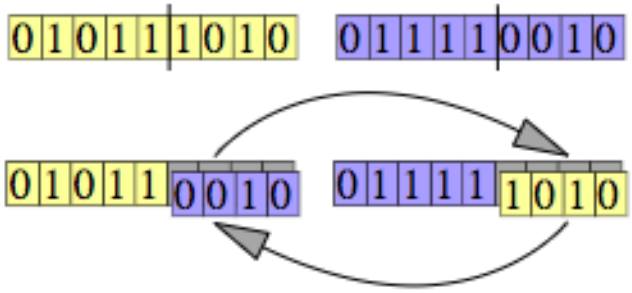

Fig. 17. Genetic Algorithm Crossover, Source: Wikimedia

In biology, animals breeds by mating and the descendant's DNA is taken from half of each genetic parents. In a GA, the same thing called "crossover" occurs at the time of breeding. One half of the father's genome, and the other half of the mother's genome are mixed together to make a new and unique child genome. The ratio of how genetic parents' genomes are split can be different each time of crossover, which results in even siblings with same parents having unique genomes.

While Genetic Algorithm is believed to potentially be used in other fields and solve some types of problems, GA's script is as simple as storing a few numbers. Doing crossover takes two numbers from parent genomes as input and calculate single number as child genome. Since the smart contract which runs the whole CryptoKitties is available on Ethereum blockchain [30], anyone have an access to the script on Etherscan.

The important idea used in CryptoKitties is that instead of ERC(Ethereum Request for Comments)-20-the most used Ethereum protocol in ICOs, ERC-721 is used [27]. On one hand ERC-20 is "fungible token", on the other hand, ERC-721 is "non-fungible token". ERC-721 is a novel Ethereum protocol standard proposed by Dieter Shirley in late 2017 on EIP(Ethereum Improvement Proposal), which makes ERC-721 tokens possible to be traded based on its uniqueness and rareness as value. Digital cats in CryptoKitties is "non-fungible token" following ERC-721 standard, which guarantees each cat being different from each other.

Merrian-Webster defines "fungible" as follows: Fungible –"being something (such as money or a commodity) of such a nature that one part or quantity may be replaced by another equal part or quantity in paying a debt or settling an account". In this case, fungibility is fundamentally a characteristic of an asset or token, that determines whether the same quantities of the same items are interchangeable during exchange. For instance, legal currencies issued by Federal Reserve are interchangeable with items with the same or less value, and currency itself can also be exchanged with the same amount of the same currency. However, one cannot buy items at store with baseball card which is traded for 10 Euro among collectors because the baseball card is non-fungible.

The simplified ERC-721 contracts example functions: name, symbol, totalSupply, balanceOf, ownerOf, approve, takeOwnership, transfer, tokenOfOwnerByIndex, and tokenMetadata [31]. To enable "non-fungible tokens" to have an unique attribution from each other, metadata containing references, like an IPFS hash or HTTP(S) link is stored in contract since storing whole data on blockchain requires extremely expensive gas(needed to execute any computation on Ethereum blockchain) [32]. References link to a program prepared outside of the blockchain and execute program to find the according data.

```
contract ERC721 {
  Required methods
function totalSupply()
 public view returns (uint256 total);
function balanceOf(address _owner)
 public view returns (uint256 balance);
function ownerOf(uint256 _tokenId)
 external view returns (address owner);
function approve(address _to,
 uint256 _tokenId) external;
function transfer(address _to,
 uint256 _tokenId) external;
function transferFrom(address _from,
 address _to, uint256 _tokenId) external;

 Events
event Transfer(address from,
 address to, uint256 tokenId);
event Approval(address owner,
 address approved, uint256 tokenId);
```

```
21
22   Optional
23  function name() public view
24   returns (string name);
25  function symbol() public view
26   returns (string symbol);
27  function tokensOfOwner(address _owner)
28   external view returns (uint256[] tokenId
29  function tokenMetadata(uint256 _tokenId,
30   string _preferredTransport)
31   public view returns (string infoUrl);
32  }
```

4) Other Use Cases: Blockchain use cases in financial industries are called "Blockchain ver. 1" [18]. In addition to these use cases in IoTs and Game Industries, there are a number of "Blochchain ver. 2" innovative use cases worth mentioning. Communication: Synereo and Reveal. Asset management: Uphold, Factom. Storage: Stroj, BigchainDB. Authentication: Block verify, OneName. Voting: Neutral Voting Bloc. Basic Income: GroupCurrency. Supplychain: Skuchain, Provenance, Bitgold, Everledger, OpenBazaar.

XI. Limitations

Blockchain technology is not a panacea. Even though there are numerous industries receiving some benefits from integrating Smart contract or other blockchain based technologies into traditional economy, ROI differs in each fields. In financial technology, IoTs and game industries, benefits of autonomous and immutable smart contract was enormous. As it is expected this technology is likely to disrupt current industries at exponentially rapid pace, regulations have not completed at enough level. Whereas security problems have been a controversial topic among developers, an asymmetric-information could be one reason which hamper correct usage of the technology in economy.

A. Technical Challenge

- New block takes time to be made Although how long depends on types of blockchain, data transaction needs about a few seconds 10 mins to be verified. Thus, blockchain technology is not suitable for applications or services which requires an instant transaction.

- TPS(transactions per second) is very low How much data of transaction can be stored in one block and how long it takes for one block to be verified leads to tps(transactions per second). Bitcoin: 7 tps, Ethereum: 20 tps, PayPal: 193 tps, Visa: 24,000 tps [33]. As it can be seen from the fig.18, the existing payment method: Visa is much faster than Bitcoin and Ethereum.

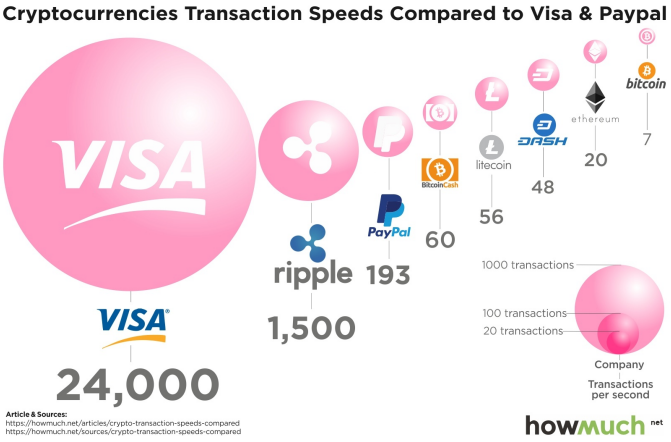

Fig. 18. Transactions per seconds compared, source:HowMuch.net

B. Regulations

Slow regulation over cryptos is also one of factors hampering blockchain adoption. According to TokenData [34], the total amount of capital raised by ICOs in January, $1,500M was a record in the crypto history. As last December and January were the months that crypto economy had drastically grown with a huge influx of funds, a great deal of attention around the world has been paid to cryptos and blockchain technology.
However, in spite of this overwhelming growth, Bitcoin.com News claims $1.36 billion worth of cryptocurrencies have been stolen by various ways including digital theft, phishing, fraud and hacking during first two months in 2018 [35].

The Chinese government is one of those countries showing strict attitudes toward cryptos. South China Morning Post reported that Chinese government blocked all websites related to cryptocurrency trading and ICOs with a purpose to swipe away any possibility of financial and political risks [36]. As many other countries realized the imminent need for legislative regulations to be formed in order to protect investors from financial risks, G20 is going to be held in Argentina on 19-20 March, to seek how participating countries call a "common response" on regulation for the phase of practical utilization [37].

C. Boundary Effect

Challenges are likely to arise at the boundary of two opposite things, for example, digital world and physical world, programmer and non-programmer, digital currency and fiat currency. Since the digital world, especially distributed ledger is distinguished by immutability, fairness through distributed consensus, transparency, these properties do not extend to fiat currency, goods, nature in the physical world. As long as the boundary of both digital and physical world exists, it can be a threshold for challenges. In the result, the use of Smart contract or any other DL technology as a part

of integration in physical economy raises risks and incentives for double spendings and block mining rewards [19].

The largest hacking theft in the crypto history happened also at the boundary between centralization and decentralization. One of the biggest crypto exchange in Japan, Coincheck announced on January 26 in 2018 that $534 million worth of XEM were stolen by unknown hackers [38]. Coincheck explained at press release in March, the hacking attack happened because one of their employees' computer was infected with malicious virus which is believed to be infected from opening an email. As a hindsight, their security measures were far from secure: customers' funds were stored in hot(online) wallet and there was no implementation of multi-signature [38].

While blockchain is such a innovative technology which will potentially disrupts almost every industry fields, people living in the incumbent industries might be hesitant about adopting something new. New technology does not bring only advantages but also emergences of unexpected flaws and security risks are possible. In the case of Smart contract and blockchain based technology in general, people's hesitancy in adoption of the technology is notably seen in the boundary between programmer and non-programmer. They demarcate an asymmetric-information threshold which hampers scaling of the technology economy wide as fast as it is hoped. By considering the proportion of coders and non-coders –the population of coders all over the world is 22 million [39]– an asymmetric information is serious. Those especially non-coders with little knowledge about DL technology has higher possibility to either overly reacting or underestimating the potential risks, regardless whether or not IT experts consider it a serious problem.

Those with little knowledge are prone to be readily affected by information media. Especially, the significant case of blockchain is hacking thefts at cryptocurrency exchange. When Coincheck announced $534M XEM theft, those people would have possibly learned that XEM's cryptography technology' security flaw was found and attacked. Consequently, they end up with biased knowledge of blockchain technology being not safe.

XII. Conclusion

As we have demonstrated, the process automation with the use of Smart contract offers business a number of attractive benefits which change incumbents of industries' problems in almost every field. Smart contract technology dramatically reduces transaction costs whereas any application can be deployed on Ethereum blockchain, then it is autonomously executed without any intervention of TTP or middleman. Furthermore, as transaction data is stored in distributed ledgers, it ensures transparency and security risks for data mutation by third party as long as nodes continues verification.

Smart contract or DL based technology has already started to be integrated into our economy with many ambitious startups emerging. Even though there are many challenges which hamper the quickest adoption of the technology, Garter's hype cycle tells us that the phase of practical utilization is incoming after blockchain experienced the phases of positive and negative hype. Adequate level of legislative regulations and public opinions against blockchain are needed for the sake of the healthy growth of the technology which is expected to disrupt many incumbent industries. After overcoming emerging challenges, the practical adoption of process automation by Smart contract is anticipated to accelarate and bring a great deal of positive influence in our economy.

References

[1] W. E. Forum, "Deep shift: Technology tipping points and societal impact," Global Agenda Council on the Future of Software and Society, Tech. Rep., September 2015.

[2] C. M. Jeremy Drane, "Pwc's 3 predictions for blockchain tech in 2016," https://www.coindesk.com/pwcs-3-predictions-blockchain-2016/, December 2015.

[3] A. Linden and J. Fenn, "Understanding gartners hype cycles," *Strategic Analysis Report No. R-20-1971. Gartner, Inc*, 2003.

[4] J. Rowley, "How ethereum became the platform of choice for ico'd digital assets," https://techcrunch.com/2017/06/08/how-ethereum-became-the-platform-of-choice-for-icod-digital-assets/, June 2017.

[5] J. Chokun, "Who accepts bitcoins as payment? list of companies, stores, shops," https://99bitcoins.com/who-accepts-bitcoins-payment-companies-stores-take-bitcoins/, March 2108.

[6] M. White, "Digitizing global trade with maersk and ibm," https://www.ibm.com/blogs/blockchain/2018/01/digitizing-global-trade-maersk-ibm/, January 2018.

[7] M. Nieto, F. Lopéz, and F. Cruz, "Performance analysis of technology using the s curve model: the case of digital signal processing (dsp) technologies," *Technovation*, vol. 18, no. 6-7, pp. 439–457, 1998.

[8] Statistica, "Number of bitcoins in circulation worldwide from 1st quarter 2011 to 4th quarter 2017 (in millions)," https://www.statista.com/statistics/247280/number-of-bitcoins-in-circulation/, 2018.

[9] M. Crosby, P. Pattanayak, S. Verma, and V. Kalyanaraman, "Blockchain technology: Beyond bitcoin," *Applied Innovation*, vol. 2, pp. 6–10, 2016.

[10] Coindesk, "How do bitcoin transactions work?" https://www.coindesk.com/information/how-do-bitcoin-transactions-work/, January 2018.

[11] A. Varshney, "Blockchain architecture," https://www.pluralsight.com/guides/software-engineering-best-practices/blockchain-architecture8o5Ztx81olTqswxO.99.

[12] G. Wood, "Ethereum: A secure decentralised generalised transaction ledger, ethereum project yellow paper," *URL https://ethereum. github. io/yellowpaper/paper. pdf*, 2014.

[13] A. Bodrova, "What are decentralized applications (dapps)," https://medium.com/ethereum-dapp-builder/what-are-decentralized-applications-dapps-ed7459a27786, December 2017.

[14] K. Lauslahti, J. Mattila, T. Seppälä *et al.*, "Smart contracts–how will blockchain technology affect contractual practices," *The Research Institute of the Finnish Economy*, 2016.

[15] F. Hawlitschek, T. Teubner, and H. Gimpel, "Understanding the sharing economy–drivers and impediments for participation in peer-to-peer rental," in *System Sciences (HICSS), 2016 49th Hawaii International Conference on*. IEEE, 2016, pp. 4782–4791.

[16] I. Constantiou, A. Marton, and V. K. Tuunainen, "Four models of sharing economy platforms." *MIS Quarterly Executive*, vol. 16, no. 4, 2017.

[17] L. Luu, D.-H. Chu, H. Olickel, P. Saxena, and A. Hobor, "Making smart contracts smarter," in *Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security*. ACM, 2016, pp. 254–269.

[18] "International survey over services which use blockchain technology," http://www.meti.go.jp/press/2016/04/20160428003/20160428003-1.pdf, Ministry of Economy, Trade and Industry.

[19] K. Kaivanto and D. Prince, "Risks and transaction costs of distributed-ledger fintech: Boundary effects and consequences," *arXiv preprint arXiv:1702.08478*, 2017.

[20] V. Pureswaran. Device democracy saving the future of the internet of things. http://www-935.ibm.com/services/us/gbs/thoughtleadership/internetofthings/. IBM.

[21] Internet of things (iot) connected devices installed base worldwide from 2015 to 2025 (in billions). https://www.statista.com/statistics/471264/iot-number-of-connected-devices-worldwide/.

[22] K. Christidis and M. Devetsikiotis, "Blockchains and smart contracts for the internet of things," *IEEE Access*, vol. 4, pp. 2292–2303, 2016.

[23] D. C. Ann Bosche. How providers can succeed in the internet of things. BainCompany.

[24] S. Tual, "Self-driving doesnÄôt mean autonomous," https://blog.slock.it/draft-self-driving-doesnt-mean-autonomous-5471d82630e4, 2017.

[25] "Global games market report 2017," https://newzoo.com/solutions/standard/market-forecasts/global-games-market-report/, 2017.

[26] Gamenation, "Using blockchain and smart contracts to solve game industry problems." *Medium*, 2017.

[27] B. Akolkar, "Cryptokitties popularity creates a new fad as more crypto collectible apps get lined up," 2017.

[28] A. Breen, "How does cryptokitties.co work?" https://medium.com/@aidobreen/how-does-cryptokitties-co-work-e5071c0abf73, 2017.

[29] kittysales. [Online]. Available: https://kittysales.co/

[30] Etherscan. [Online]. Available: https://etherscan.io/address/0x06012c8cf97bead5deae237070f9587f8e7a266d#code

[31] [Online]. Available: https://github.com/dete

[32] G. Nash, "The anatomy of erc721," https://medium.com/crypto-currently/the-anatomy-of-erc721-e9db77abfc24.

[33] Raul, "Transactions speeds: How do cryptocurrencies stack up to visa or paypal?" https://howmuch.net/articles/crypto-transaction-speeds-compared, January 2018.

[34] "Capital raised by icos," https://mailchi.mp/tokendata/tokendata-weekly-newsletter-289971, January 2018.

[35] K. Sedgwick, "9 million a day is lost in cryptocurrency scams," https://news.bitcoin.com/9-million-day-lost-cryptocurrency-scams/, 2018.

[36] X. Yu, "China to stamp out cryptocurrency trading completely with ban on foreign platforms," http://www.scmp.com/business/banking-finance/article/2132009/china-stamp-out-cryptocurrency-trading-completely-ban, February 2018.

[37] G. Argentina, "First meeting of finance ministers and central bank governors of 2018."

[38] J. Young, "530 mln in xem stolen from coincheck can be traced, nem team confirms," https://cointelegraph.com/news/530-million-in-xem-stolen-from-coincheck-can-be-traced-nem-team-confirms, 2018.

[39] E. D. Corporation, "Global developer population and demographic study 2017 vol. 2," https://evansdata.com/reports/viewRelease.php?reportID=9, 2017.