FinTech

Article

Conditional Token: A New Model to Supply Chain Finance by Using Smart Contract in Public Blockchain

Che-Pin Chen [1,*] , Kai-Wen Huang [2] and Yung-Chi Kuo [2]

1   Department of Risk Management and Insurance, National Cheng Chi University, Taipei 116011, Taiwan
2   Department of Management Information Systems, National Cheng Chi University, Taipei 116011, Taiwan
*   Correspondence: actuary@ms59.url.com.tw

Abstract: This paper defines Conditional Token (CT) as the token with specific conditions and proposes the use functions for its operations in smart contract so that it can be deployed at the public blockchain. If CTs were exchanged to/equivalent to fiat currency once then all conditions are realized, that is, the required performances and obligations/rights are agreed upon. In use, the obligation-type CT can be used as a divisible mortgage or be used as a representation of accounts receivable, accounts payable and vouchers as it is used in accounting. While the rights-type CT can be used as divisible fixed-income bonds or as an investment vehicle. Integrate both types of CTs with a matching methodology can thus be used in any kind of peer-to-peer (P2P) system of the decentralized finance, such as crowdfunding and P2P lending. This paper thus applying this new model to solve the complex issues of supply chain finance. For feasibility, this study concludes CT is the "Verdinglichung Obligatorischer Rechte", and CTs are better than the current corporate loans in terms of cost and benefits. In addition, it is capable of transferring risk to other investors. In terms of implementation, this paper proposes a system framework and has completed a proof of concept of the system.

Keywords: supply chain finance; smart contract; DeFi; conditional token; verdinglichung obligatorischer rechte

---

1. Introduction

The supply chain of a product refers to all activities between initially sourcing raw materials to the sale of goods to customers, and includes equipment, production, inventory, sales, after-sales service and other matters. Even while companies in the supply chain are accepting orders or conducting operations, there might be funding gaps in their respective production and sales operations, thus forming capital needs. However, most upstream and downstream manufacturers are small and medium-sized enterprises, and it is not easy for them to get loans from banks. The issuer provides financing for the guarantor, resulting in a system of supply chain finance (SCF).

However, SCF is easier said than done. It's scope is generally divided into cash flow management, financial instruments, and buyer-to-pay solutions [1,2]. And it's complexity being clear from twelve categories of financial instruments [3], and from the horizontal and vertical relationships in the supply chain [4] (p. 5), as shown in Figure 1.

The vertical relationship is also portrayed as N + 1 + N. The first N is the upstream manufacturer, 1 is the core manufacturer, and the second N is downstream manufacturer or distribution channels. The types of SCF in the vertical relationship include receivables purchase agreements, loans and prepayments [5]. There are also three other types according to the specific supply chain process: purchase order financing, inventory financing, and accounts receivable financing after delivery happened [3]. In addition, letters of credit and bank guarantees are used in international trade which are classified as horizontal relationships, and which can improve the enabling framework of SCF [5]. The vertical and

horizontal relationship increases the difficulty for financial institutions to clarify the relationship with core enterprises in financing, and reduces the willingness of core enterprises to guarantee supply chain manufacturers, reducing the overall effectiveness of SCF. The solution proposed in this study, is applicable to the relationships as shown in Figure 1, and covers the cash flow management of the supply chain.
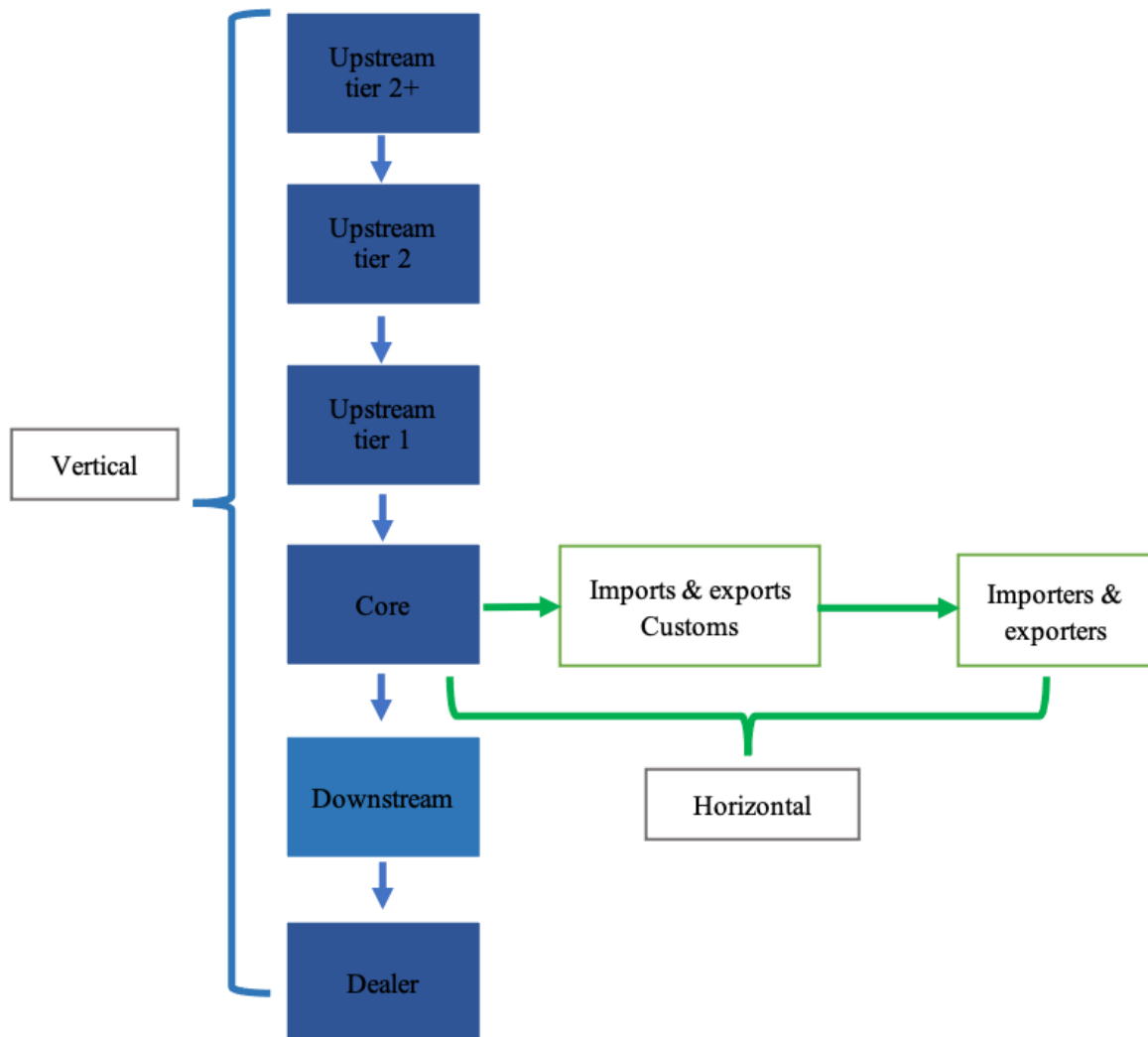

Figure 1. The scope of SCF standing position at "Core".

Figure 1 shows that there are many participants in SCF. Therefore, the financial industry expects core companies to act as guarantor and expects them to reduce the default risk. However, the difficulty of verifying the transfer of claims, easily leads to fraudulent claims and an increase in social costs. In addition, insufficient information also creates obstacles for legal compliance, accounting and transaction costs.

To improve the transparency of information, some scholars have begun to propose solutions by means of blockchain and securitization [6–8]. Blockchain technology has become famous since it was first applied to Bitcoin transactions in 2013. It is characterized by the fact that the recorded peer-to-peer information cannot be forged and tampered with, and is easy to trace [9]. Ethereum proposed the use of smart contracts, which provides a programming language to write programs. After compiling and deploying virtual machines on each node, the nodes of the blockchain can be used to execute the functions in the program or change the state of the blockchain, with that creating an open ledger [10].

Theoretically, through the use of game theory, Dong [11] argues that blockchain technology can improve the visibility of deep-tier financing (higher than the tier 2 in Figure 1).

Choi [12] applied the Nash bargain model to prove that blockchain-enabled supply chains have lower operational risks than traditional supply chains, and that suppliers will benefit from blockchain-based supply chain financing. For practical use, Hofmann et al. [6] studied blockchain-oriented reverse financing mechanisms (Reverse Securitization), Lahkani [13] combined the blockchain with the e-commerce supply chain to achieve a simplified supply chain, Chen [14] provided a framework for implementing SCF for the automotive industry, Tsai et al. [15] studies the use of elliptic curve cryptography as a security mechanism of SCF in the agricultural industry, which is actually a blockchain verification methodology, lastly, Du et al. [16] proposed a blockchain-based SCF platform system.

Supervisory agencies, international institutions, and scholars study affirm that the application of blockchain in supply chains make sharing of information possible, and that it can be more efficient for parties new in the financing industry [17–19], can facilitate international payments [20], automate transactions, duplicate financing and secret protection [21]. In practice, FNCONN Financial [22] issues "Fujin" as a payment commitment and combined with Chain Finance issues "FP" Tokens in the form of a consortium chain as debt certificates (credit asset certificates).

However, the decentralized nature of the blockchain, coupled with the application of internet, reduces the intermediary role of banks. Currently, to assist small enterprises in financing, peer-to-peer/person-to-person (P2P) lending through internet or e-commerce platforms, has been put to use in SCF, for example: Ant Financial Services, Zopa UK, The Receivables Exchange USA etc. However, as unspecified persons are being served, identity verification, personal information protection or money laundering prevention and control are all the difficult issues or even obstacles, resulting in a dilemma between supervision on one side and inclusiveness on the other side [23]. As a matter of fact, it is confirmed that reducing the quality of financing does bring a lot of drawbacks to the P2P financial lending platform [24]. With that the issue of legal compliance begins [25,26].

Therefore, this study believes that the current use of blockchain for SCF still requires three issues that need to be solved. First, current research [13,14,17] focuses on the passive nature of using the blockchain in log records, which fails to use the advantages of smart contract programs to issue and operate tokens. Second, the application of blockchain in SCF mostly adopts permission-based (or membership-based) consortium chains [14,16,27], which limits the inclusiveness, and the credibility of the system because of the limited number of nodes. Third, is the feasibility of blockchain in SCF, including its cost-effectiveness and legal nature.

In addition, asset tokenization becomes popular recently, either physical assets or intangible assets. Tokenization is a technology that allows the representation of real-world assets in a digital form on a blockchain network [28]. These tokens can then be traded on a blockchain network, allowing for improved liquidity, fractional ownership, and reduced transaction costs. To achieve this, tokenization and operations require the use of smart contracts mentioned in the above, such as ERC-20 [29] or ERC-721 [30]. In this article, we tokenized the obligation conditions of debt, and the right conditions of investment which is called conditional token. The conditional tokens are used to solve the issues in SCF.

Concluding the above, conditional token is defined mathematically and issued and driven by operations proposed in this research. All can be written down in smart contracts and deployed on the public blockchain, not only to record the specific conditions of using these tokens, but also for the purpose of transferring among different tiers of manufacturers as to achieve N + 1 + N, and can even act as account receivable or accounts payable in line with common accounting principles. In addition, the public blockchain is the most credible, and can therefore improve credibility of users in its P2P architecture thus achieve a higher inclusiveness. For practical purposes, this article discusses the legal nature of CT, as the "verdinglichung obligatorischer rechte", introduces a risk management tool, and proofs that a system which uses CT is more cost-effective than current banking practices.

For illustration and implementation purposes, the related smart contracts are deployed on the testnet as shown in the Appendix A, the system has finished a proof of concept, as well as participated in the competition in which it achieved good results (the First prize of

Academia and Industry Collaboration, ITeam PRI-08, International ICT Innovative Service Awards, https://innoserve.tca.org.tw/en/guidelines.aspx, access date: 1 March 2023).

The remainder of this article is organized as follows, Section 2 introduces and reviews the articles of blockchain, smart contract, and tokenization. Section 3 explains the issuance, conditional and operational functions of CT's. Section 4 describes how to apply CT's for the purpose of SCF. Section 5 analyzes the feasibility of practical operations, while the conclusions and recommendations are being given in the final section.

2. Blockchain and Smart Contract

2.1. Blockchain

The original concept of blockchain is from the technique of cryptographic timestamp to prevent data tampering and unreliable timestamps proposed by Stuart Haber, W. Scott Stornetta, and Dave Bayer in 1990's [31,32].

Decentralized blockchain was first introduced in 2008 and is well known as "Bitcoin" [9]. Bitcoin solves the double-spending problem without the need of a trusted authority or central server by applying peer-to-peer network or "decentralization".

In general, a blockchain is a distributed ledger contains blocks which are securely linked via cryptographic hashes of the previous block, including a timestamp, and transaction data; while the ledger information is recorded in the block [33]. In addition, there are three types of blockchain systems, from the most decentralization to the least would be public blockchain, consortium blockchain, and private blockchain [10]. Zheng et al. (2018) [34] compare them from different perspectives.

Trust is a basis of transactions, the features of blockchain, decentralization, transparency, immutability, and traceability, make them more secure, tamper proof, more cost effectives and increase efficiency [16,18,34]. Therefore, a large number of applications are proposed based on blockchain, including Internet of Things, finance, healthcare, supply chain, and so on [12,35–41].

P2P network in public blockchain is called mainnet, the transaction cost is high in mainnet, and all of the transactions cannot be retreated. Testnet is thus developed in the ecosystem of blockchain such as Ethereum. Testnet is a type of blockchain network, it is a simulated blockchain network that mimics the functionalities of a mainnet, but without any real value attached to it. Testnets are designed to be a safe environment for developers and users to test new ideas and features without risking real assets or disrupting the main blockchain network [42]. In this article, the Goerli testnet is used for the implementation, it was the first proof-of-authority cross-client testnet, synching Parity Ethereum, Geth, Nethermind, Hyperledger Besu (formerly Pantheon), and EthereumJS [43], the proposal is in [44].

2.2. Smart Contract

The advent of blockchain technology has paved the way for innovative solutions, and the smart contracts deploying in the blockchain are one of the most promising. Szabo (1996) [45] first proposes the concept of smart contract which executes the terms of a contract by a computerized transaction protocol. The turning-complete programming language was proposed to run user-defined functions on blockchain to operate "contracts" [10,33]. We thus define smart contracts are self-executing digital programs that run on a blockchain; while execute automatically when certain conditions are met. Compares to blockchain, smart contracts solve the issues of lack of flexibility, time exhausting transaction, high computing power and energy consumption, and privacy transactions requirement. Therefore, smart contracts are self-executing, tamper-proof, and enforceable, and can be used in a wide range of applications, from supply chain management to financial services [46,47].

There are several high-level programming languages that are used to write smart contracts on different blockchain platforms such as Solidity [48] which is the most popular. The codes of those languages can then be compiled into bytecodes to be run on virtual machine. Ethereum is currently the most popular platform for developing smart contracts, as well

as the EVM, Ethereum Virtual Machine, hence it is called "Blockchain" or "Distributed Ledger" 2.0 [33] and [49] (pp. 105–118). In addition to Ethereum, there are some other platforms which can be utilized to develop smart contracts, such as Michelson in Tezos blockchain, Bamboo in Hedera Hashgraph blockchain, etc. which beyond the scope of this article.

When a function of smart contract is executed, if it stores data on the blockchain, then the store is equivalent to send a transaction to blockchain, and will change the status of distributed ledger, so that the data cannot be modified or deleted. For example, transferring of a certain amount of cryptocurrency by a transfer function.

Smart contracts offer a number of benefits over traditional legal agreements. The feature of self-executing eliminates the need for intermediaries, such as lawyers, notaries or even financial companies, and reduces the time and cost of executing contracts [50] (pp. 16–18). Smart contracts are also tamper-proof, meaning that once they are executed, they cannot be altered. This makes them more secure than traditional contracts, which are vulnerable to tampering and fraud.

Smart contracts have a wide range of applications in academic study, 64% of all research related to smart contracts [51]. For supply chain, it applies in different industry widely such as construction [52] and agri-food [41], in the process control [53–55], as well as for financial services [6,12,14]. In supply chain management, smart contracts can be used to track the movement of goods from the manufacturer to the retailer, a compressive and systematic reviews by [19]. In financial services, smart contracts can be used to execute complex financial transactions automatically, such as the transfer of assets from one party to another, recent reviews by [56], and [57] for supply chain finance.

Despite their many benefits, smart contracts face several challenges. One of the main challenges is the lack of standardization. There are many different programming languages and blockchain platforms, which can make it difficult to develop and deploy smart contracts. However, recent research shows the different platform suffering common risk [58]. Another challenge is the complexity of smart contracts. Smart contracts can be difficult to write and understand, which can make them vulnerable to errors and bugs [47]. In this article, we import standard contracts from OpenZeppelin. It provides security products to build, automate, and operate decentralized applications [59]) to several smart contracts for different targets which are written by Solidity.

2.3. Tokenization

Tokenization is a technology that allows the representation of real-world assets in a digital form on a blockchain network. Treiblmaier (2021) [28] classifies it to utility tokens, payment tokens, and investment tokens. This enables the digital tokens that can show the bestowing of a right to their owners, exchange value, use of a platform or services (toll), enrich user actions (function), enable frictionless transactions (currency), or receive a fair redistribution of value (earnings) on a trustful platform, without the need for intermediaries [60]. Tokenization has been gaining popularity in recent years, with many companies and organizations exploring its potential for various use cases.

These tokens can then be traded on a blockchain network, allowing for improved liquidity, fractional ownership, and reduced transaction costs. To achieve this, tokenization and operations require the use of smart contracts mentioned in the above, such as ERC-20 [29] or ERC-721 [30].

Recently, there are various applications of tokenization, including the tokenization of materials among supply chain [61], financial assets [62], real estate [63,64], art [65], and intellectual property [66]. One of the most popular use cases for tokenization is the creation of security tokens [63]. Security tokens have gained traction in the financial industry, as they offer a more efficient way to raise capital and trade assets. Token economy via tokenization is coming.

Tokenization is a powerful technology that has the potential to revolutionize various industries. Its benefits include improved liquidity, fractional ownership, and reduced

transaction costs. While there are challenges associated with tokenization, such as legal and regulatory issues, cybersecurity and business ownership [67].

In this article, we tokenize the financing conditions as debt and investment conditions as security, issue the obligation-type conditional tokens and the right-type conditional tokens, respectively.

3. Conditional Tokens

This section will explain the definition of CT's in both a mathematical and descriptive manner, as well as through the operation (aka functions) and operation processes that are being used. The smart contract is deployed in the blockchain by using an unique address, all of its functions will be driven by the system's backend and the results can be saved in a transparent manner on the blockchain.

3.1. Definitions

Define C_i is the ith condition $C = C_1, C_2, \ldots$ is the set of all "conditions", and Ξ is the power set of C.

Definition 1. *An activated function $\delta : C \rightarrow \{0, 1\}$, we called C_i is activated or achieved if $\delta(C_i) = 1$, and (E, δ) is called a conditional token (CT), denoted by $CT \equiv (E, \delta)$.*

In computer, CT can be programmed as the structure type as in Table 1.

Table 1. Example of CT programmed by Solidity.

```
struct CT{
    address issurer;//issuer's address
    const par;//par value of CT
    uint256 interest;//condtion1
    uint256 date;//condition2
    }
```

In addition, define $\delta(\varnothing) \equiv 1$, for the purpose of completeness, and $X \equiv (\varnothing, \delta)$ is called currency token which is equivalent to fiat currency.

CT is a struct or vector type, it cannot be operated or measured by the current standard such as ERC-20 [29] or ERC-721 [30], thus operations or functions are requires to measure and operate CT. By the way, an address is required for the ownership of CT. The address in blockchain represents the account as well as the public key of the token holder, let $\mathcal{A}$ be the set of all the addresses in blockchain.

Definition 2. $T : \mathcal{A} \times \Xi \rightarrow \mathbb{R}$ *and $T(a, E)$ is the total number of $CT \equiv (E, \delta)$ that belongs to address a.*

In this article, a time index $T_t(a, E)$ is used, to specify different times if needed. Besides, $T(a, \varnothing)$ is the total number of currency token or of fiat-backed currency of account a. For convenience, we define $X(a) \equiv T(a, \varnothing)$ for currency token $X \equiv (\varnothing, \delta)$.

In practice, two types of CT are used, one is an obligation-type, the other is a right-type. For the obligation-type, there are two processes of issuing (aka tokenized collateral), with the first process locking-in collateral. It is very convenient for crypto collateral, however IoT devices are required for physical collateral, which is not address in this paper. Then, as the second process, CT's backed by collaterals are minted.

Figure 2 shows the process of issuing the obligation-type CT, the demand (issuing) side requests to set mortgage as collateral tokens and provides the specific loan conditions, where the collateral tokens are held in smart contracts (①). If the collateral tokens are created by a third-party such as ETH, or need to be evaluated by a third party, then verification from a third party is required (②). After the verification is completed (③), the smart contract issues/mints a predetermined number of CT's (④).
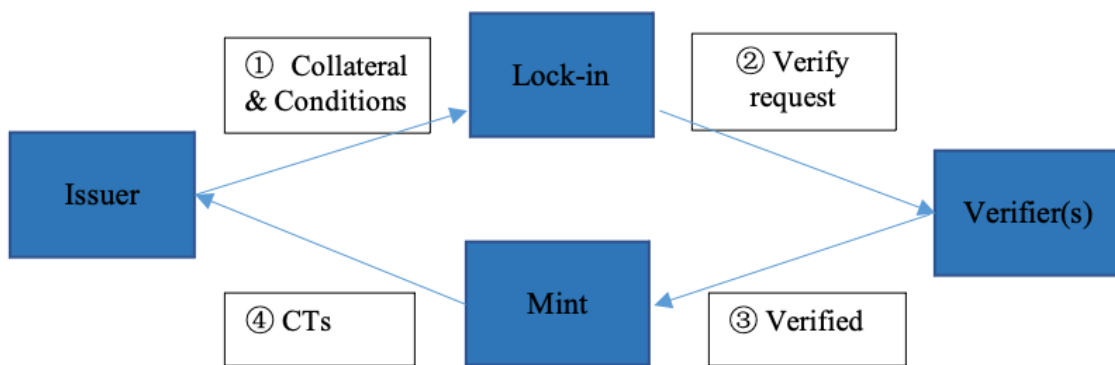
Figure 2. Obligatory-type CT.

The above-mentioned custody mechanism is to transfer the collateral to the address deployed in the smart contract, which is equivalent to "locking in" the collateral as a trust mechanism. The protocol of "lock" or "unlock" can apply or revise ERC-1132 [68], however, this will not be addressed in this article.

Similar to the obligation-type, there are two processes of issuing right-type CT, depositing fiat currency to receive account tokens (aka collateral in obligation-type), then transfer account tokens to the "lock-in" address, followed by deciding the "rights". The first process can be done by an application program interface (API) from a third-party payment or bank service. The smart contract will issue the account tokens to represent the fiat currency after a successful deposit.

For the second process, the supply side transfers account tokens (①), locks them into the smart contract, then applies the mint function of smart contract to issue the conditional tokens (②) according to Figure 3. The operations are similar to the obligation-type CT.
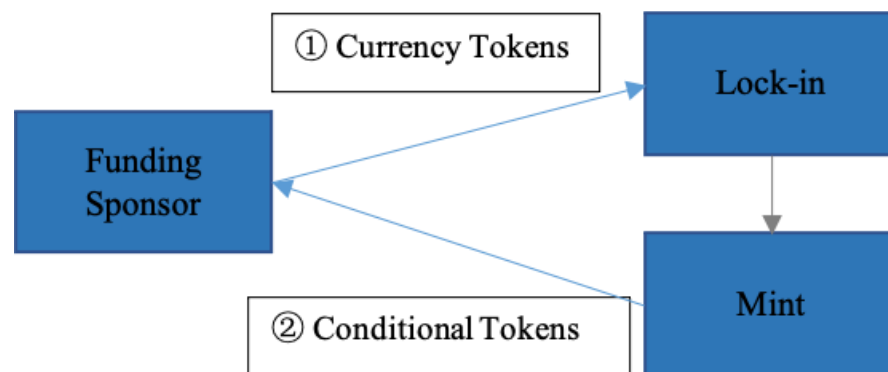
Figure 3. Right-type CT.

Definition 3. *Mint function μ of account a, increase the amount of CT's from the CT balance of a with the same conditions as E. That is,* $T_t(a, E) = \mu \circ T_{t-1}(a, E) = T_{t-1}(a, E) + \mu(a, E)$.

The mint function is used to issuing of CT's. If the number of m is minted for address a of CT, then add amount m to $T(a, E)$, the use case for "mint" are illustrated in the Appendix A.

Based on the definitions and the mint operation mentioned above, CT's are minted (or issued) by a smart contract, while the (either obligations or rights) conditions are confirmed and recorded on the blockchain. When the CT is used, the agreed conditions are achieved. In other words, the use of CT needs to fulfill (exercise) the agreed obligations (rights) at the same time.

Simply put, the obligation-type CT is divided into three types: inventory CT's (inventory backed), pledge CT's (backed by something other than inventory) and margin CT's (cash backed). The right-type CT's are called investment CT's.

3.2. Conditions

The scope of conditions includes, but is not limited to, interest rate, term, and restrictions on transfer manufacturers, donation criteria, etc., and are determined by the issuer, fund providers based on the funding purposes or are based on market conditions.

Naturally, because conditions must be fulfilled to be valid, there is no need to assume conditional responsibility until they are fulfilled. Thus, for interest rate conditions, the obligation-type is a divisible Line of Credit Mortgages (detailed in Section 6.1), or a CT equal to a mortgage with a one-unit limit. The maximum amount of mortgage rights is the number of CT's issued after the mortgage is created.

Issuers who transfer CT's to upstream or downstream manufacturers can change the "conditions" or "terms" according to their own needs and can provide different conditions to different manufacturers. In addition, they can also limit the number of times the conditions can be transferred.

For the right-type CT, the fund providers can decide on a certain set of conditions, such as the minimum rate of return, and determine the conditions once more through matching, or directly select and agree on the conditions in advance. Similarly, the rights cannot be exercised until the conditions are fulfilled.

Therefore, the conditions can be an investment tranche with different returns, with each CT acting like a divisible fixed-income bond.

3.3. Operations

Besides minting, we will define more operations used for CT, including exchanging, transferring, modification, and burning.

Operations are the functions in smart contract which changes the state of blockchain.

Definition 4. *The exchange function Π is an operator that projects from Ξ to currency token X, so that $\Pi \circ \Delta T(a, E) = -\beta \Delta X(a) \delta(E)$, where β is the exchange rate for one CT to currency token X, and $\Delta T(a, E) = T_{t-1}(a, E) - T_t(a, E)$, $\Delta X(a) = X_{t-1}(a) - X_t(a)$.*

It means if CT exchanges to X, then E is activated, meaning, all of the conditions in E are achieved.

Definition 5. *The transfer function σ, so that $\sigma \circ \Delta T(a, E) = -\Delta T(b, E)$, transfer amount $\Delta T(a, E)$ of CT's from address a to address b.*

Definition 6. *The modification function $\Psi_i(E) = E'$, where $E = \{C_1, C_2, \ldots, C_n\}$, $E' = \{C_1, C_2, \ldots, C_{i-1}, C'_i, C_{i+1}, \ldots C_n\}$.*

After modifying account a, $CT(a, E)$ decreases, and $CT(a, E')$ increases.

Definition 7. *The burn function v of account a decreases the amounts of CT from the CT balance of a with the respective condition E. That is, $T_t(a, E) = v \circ T_{t-1}(a, E) = T_{t-1}(a, E) - v(a, E)$. All the functions or operations are programmed in the programming language Solidity and represent the smart contract* [10]*.*

Summarize the functions as in Table 2.

Table 2. The functions/operations of CT.

Functions	Inputs	Descriptions
Balance T	a	The balance of CT for address a
Mint μ	a, m, E	Mint amount m of CT with condition set E to the address a
Exchange Π	a, β, m	Exchange amount m of CT to amount βm currency token X for address a
Transfer σ	a, b, m	Transfer amount m of CT from address a to address b
Modification Ψ_i	a, E, C'_i	Modify the ith condition C_i to C'_i in condition set E for address a
Burn v	a, m	Burn amount m of CT from address a

4. The Application of CT to SCF

SCF is divided into three categories: loan or advance payments, purchase of receivables, and bank payment obligations [5].

The third category is mainly for enabling the SCF, one of the management processes, and is not within the scope of this paper. This article will apply CT's to the first type of accounts receivable (aka factoring and forfaiting), and the second type of advance funding and inventory financing, with purchase order and distributor financing being part of advance funding.

4.1. Advance Funding

Traditionally, the buyer pays in advance for the goods, however has not yet received the goods for sale or needs an early planning of its sales, resulting in a funding gap, therefore, it secures it financing by using "future trade receivables" as guarantee (Forum, 2016), which is called channel financing in case the buyer is a channel distributor. In practice, in addition to future trade receivables, it also includes the right to take future delivery of goods, or the bill of lading certificate when the goods are delivered, or picking up the goods in a third-party warehouse, and use the sales receipt to first repay the financing service (see Figure 4).
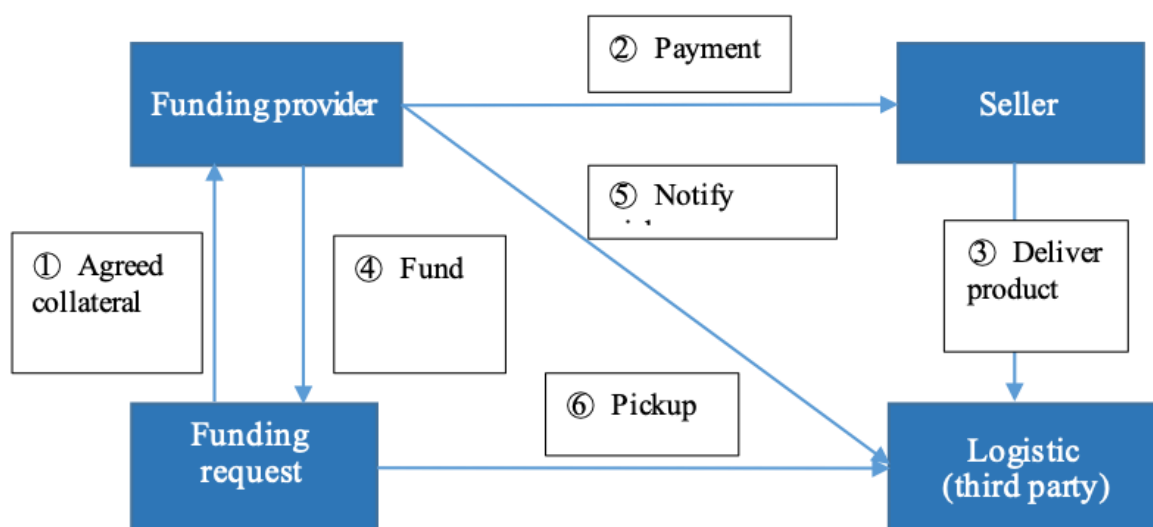
Figure 4. Advanced-based finance.

On the other hand, if the seller receives a purchase order for goods, but has not yet received cash before the product or delivery is finished, the situation results in a gap in raw material funds, leading to the purchase order being similar to the "future trade receivables" as described above (Forum, 2016). Such a situation is called purchase order or pre-shipment financing.

Regarding channel financing, in addition to risks in repayment arising due to poor sales performance, if the buyer were to be the core company, and already collateralized its inventory, then the downstream dealer can no longer obtain financing through using the same collateral.

Purchase order financing is based on the demand for funds in the entire process from placing an order until payment is received. The "future trade revenue " of the order is used as the source for loan repayment. However, because there are many upstream levels involved (Figure 1), transmission or passing on of information limited, resulting in the highest risk in this way of financing, which presents an excellent opportunity for CT's. The basic architecture is shown in Figure 5.
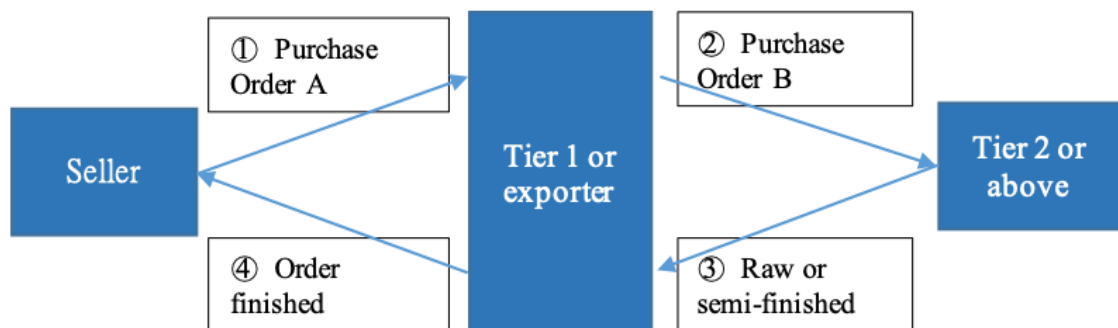
Figure 5. Purchase order.

If CT's are used, no matter whether for channel or purchase order financing, they can replace the collateral agreed upon by both parties as shown in Figure 4, because we assume core company has proof of credits already. That is, mint CTs form core company to funding requester a, and then transfer them to another company b. The operation will be $\sigma \circ (b, \mu(a, m))$, where m is amount of CT.

For channel financing, the CT is provided at the same time the sale happens, helping the dealer solve the funding problem caused by repeated pledges. For purchase order financing, the deposit can be either cash or currency tokens. Manufacturers who accept the CT can exchange it for capital, or transfer it to other manufacturers, basically solve and mitigate the risk of multi-level information transmission of purchase order financing in a system as portrayed by Figure 1. In addition, it is also possible to reset the conditions and limitations of manufacturers, such as: interest rate or term adjustment, included companies, or limit the number of suppliers, etc.

The processes and the operation algorithm of advance financing are listed as Algorithm 1.

Algorithm 1 The processes and the operations of advance financing

1. Input core company address a, then calls mint function $\mu(a, m)$.
2. Input channel address b for channel financing, or input manufacturer address b for purchase order financing, then the core company calls transfer function $\sigma(b, n)$ to transfer n CTs to address b, where $m \geq n$ (when placing purchase order to b, for purchase order financing)
3. Address b can either exchange CTs to currency or transfer them to another company.

   If exchange is true, then calls exchange function Π.

   If transfer is true, then input manufacturer address c, and calls modification function Ψ_i to modify the required conditions, and then call $\sigma(c, k)$ to transfer k CTs to address c, where $n \geq k$.

Algorithm 2 The processes and the operations of inventory financing

---

1. Input collateral company address *a*, after validate the inventory, then calls mint function $\mu(a, m)$.
2. Address *a* can either exchange CTs to currency or transfer them to another company address *c*:

   If exchange is true, then calls exchange function Π
   If transfer is true, then input address *c*, and calls modification function Ψ_i to modify the required conditions, and then call $\sigma(c, k)$ to transfer *k* CTs to address *c*, where $n \geq k$.

4.2. Purchase of Account Receivable

After upstream companies (or exporters), being the sellers, complete the order and ship the goods to the core company (or importer), while the buyer did not pay upon delivery, it will create accounts receivable (AR) that are received by the seller. The upstream manufacturers (or exporters) can use this AR for financing by selling all or a part of it to a finance provider [5], as shown in Figure 6.
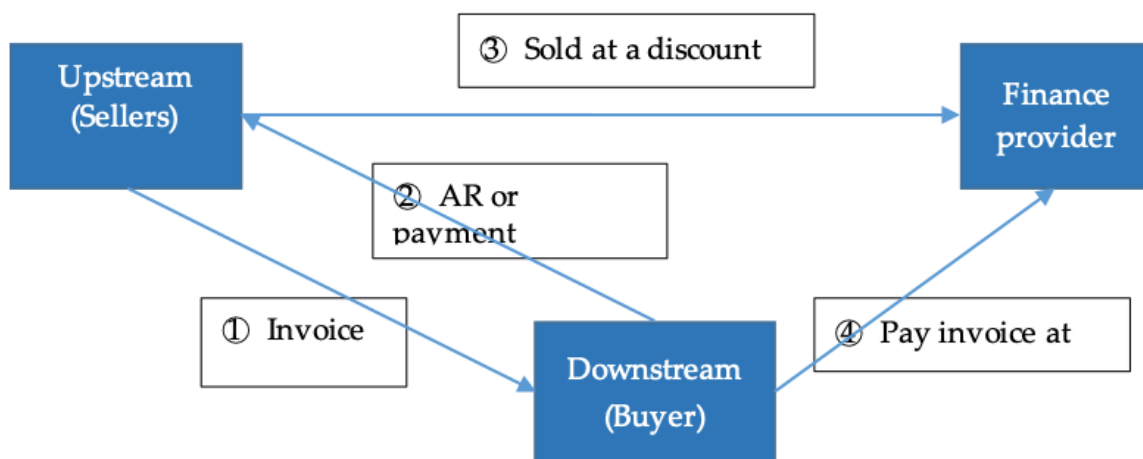
Figure 6. Accounts receivable.

If AR is sold directly, it is called factoring, with the fund provider being called factor. If the buyer issues a payment instrument such as a post-dated check to the seller, which the seller in turn sells to a fund provider, it is called forfaiting, with the fund provider being called forfeiter.

AR financing discounts short-term financing, which accountant can easily hide. It will be even more difficult to verify if second-tier or above manufacturers are involved. In practice, there are cases where the accountant letter is incorrect or even appears to be a scam (98-Jin-3, Taiwan Shilin District Court 2015, after line 137).

Figure 7 shows the structure in which CT's act as accounts receivable. The process is similar to Figure 6, and the algorithm is in Algorithm 3. The difference is that the CT's are used as accounts receivable or as a payment instrument of the buyer, so that the sellers (upstream) can sell the CT's to fund providers. Sellers can also use the tokens as AR or payment instruments when dealing with higher tier manufacturers, and can modify the conditions of the CT to reflect time value (discount) and the respective expiration date, waiting until expiration date, or to show the CT to claim payment for the goods. The transfer process of the CT will be recorded in the blockchain where anyone can verify it and thus know whether the information is correct or not.

Algorithm 3 The processes and the operations of purchase account receivables

---

1. Company address *a*, check the CT amounts, calls balance function $T(a)$.
2. Address a want to sell m CTs to address b, where $m \leq T(a)$ then

   address *b* transfer *m* currency token X to address *a*.
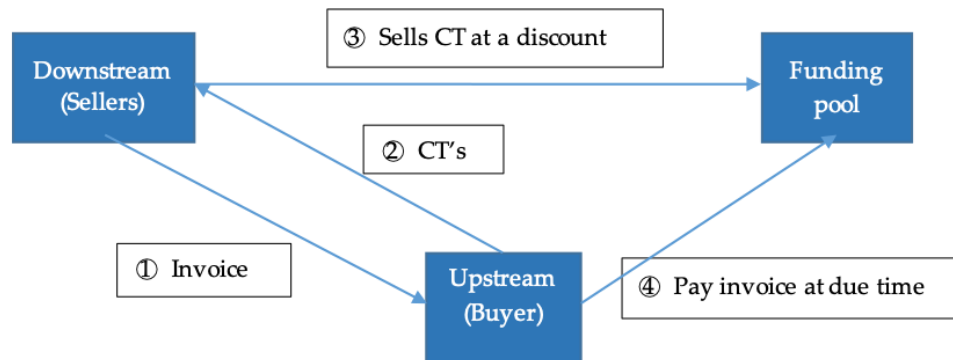   address *a* call $\sigma(b, m)$ to transfer *k* CTs to address *b*.

Figure 7. CT as the accounts receivable.

Here, for the upstream manufacturers, the conditional tokens are equivalent to the traditional bank checks for AR, and thus can be used as a proxy for AR, while core company can the conditional tokens as proxy for accounts payable. The discount price is part of the conditions. Even so, it still needs to meet accounting principles, however this is beyond the scope of this study.

4.3. Inventory Financing

Inventory financing uses inventory of raw materials, semi-finished, and finished goods as collateral to obtain capital [5]. The use of blockchain technology can reduce the risk of double loans or goods sold [16].

The CT is obtained by using the inventory as a collateral, and it requires to be verified and evaluated before issuing as in Figure 2. Because information of the CT is recorded publicly, it is easy to do verification. The core company can use the CT for itself or transfer the CT to other companies. In practice, it can be transferred to upstream manufacturers as accounts receivable (Figure 7), or provided to downstream manufacturers, such as distributors (Figure 5). Except using inventory as the collateral, the remaining operations are similar to Algorithm 1, and list in the Algorithm 2.

4.4. Peer-to-Peer/Person-to-Person (P2P) Financing

In P2P financing, it requires to distinguish obligation-type and right-type CTs, uses of the indices denoted by *o* and *r*, respectively.

CT_o's holders want exchange currency from the currency token holders by P2P financing and the investors deposit currency token X to swap CT_r for condition E_r. The relationship is in Figure 8 and the algorithm as below Algorithm 4:
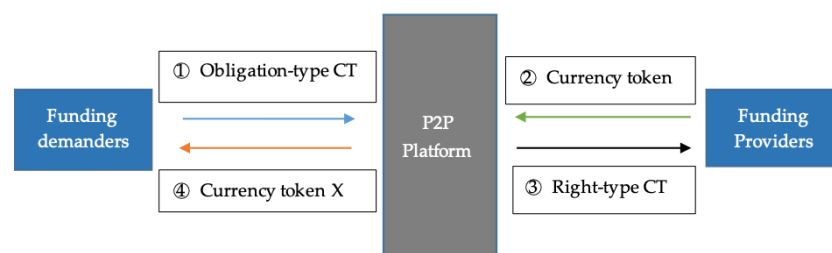
Figure 8. P2P Financing.

Algorithm 4 The processes and the operations of P2P financing

---

1. CT_o's holders address $a_1, a_2, \ldots, a_N$, call $\sigma_o(b, m_i)$ transfer their CTs to platform address b, where m_i is the transfer amount of address a_i, for $i = 1, 2, \ldots, N$
2. Platform revised condition E_o of CT_o, and shows the investment terms E_r conditions of listed CT_os
3. Investors address $c_1, c_2, \ldots, c_M$ deposit cash or transfer currency token $X(c_i)$ for $i = 1, 2, \ldots, M$
4. Investor $c_j, j \in J$ selects CT_o of c_i, and will provide k_j of currency token to fund $c_{i,}$, where J is the investors who selects c_i, $c_j, j \in J$ transfer k_j of X to platform, and platform calls mint $\mu_r\left(b, k_j\right)$, $k_j \leq m_i$ with condition E_r, then call $\sigma_r\left(c_j, k_j\right)$ to transfer CT_rs to investor j.
5. If $\sum_{j \in J} T_r\left(c_j, k_j\right) = T_o(a_i)$, the platform transfers m_i of X to address a_i.
6. Payment from a_i will exchange to currency tokens and then transfer to $c_j, j \in J$, based on the E_r.
7. Redeem from a_i, the platform calls burn $v\left(c_j, k_j\right)$, $\forall j \in J$, and return k_j of currency token to c_j

---

5. System Design and Implementation

Some research proposes a SCF system based on the consortium chain [14,16]. This study concludes that the functions of the front-end of the SCF system include user interaction, Internet of Things interaction and third-party control functions. The back-end services include data management functions and blockchain management functions. The latter being the functions of a smart contract in which the interactive function of the IoT is used as a sensing device to convert physical collateral into crypto collateral or crypto tokenized assets. This article will not cover this topic in depth. The benefits of the public blockchain proposed in this study, is that no consortium chain [14] and consensus agreement system [16] needs to be build.

5.1. SCF System

Below figure depicts the structure of SCF system. It could be hierarchically divided into six layers as shown in Figure 9, and each layer makes full use of the functionality and module provided by its below layer. The infrastructure layer utilized the platform as a service (PaaS) which is provided by a third party provider, such as AWS EB (the Elastic Beanstalk of Amazon Web Service) which is used in this article. It makes the most of capacity provisioning, load balancing, scaling, and application health monitoring. On top of the infrastructure layer, we integrate multiple data sources in the data management layer. The business management system provides data flow regarding supply chain transaction records, the database of this system defines the interface and the relation between data received from management systems and user input, and the blockchain serves as authenticity and validity of information flow. To manage the data, we suggest IoT devices or Application Interfaces (APIs) in the API layer to communicate between corporate's Enterprise Resource Planning (ERP) system and the blockchain, where ERP provides the information of collateral. With the manageable data flow, the services are proved by a series of smart contracts which are deployed in the public chain in the layer of smart contract functionality. Risk control layer supervises the financing activities, and delivers a set of risk management methods.

The website of SCF system is under testing, and the examples of operations are illustrated in Appendix A [69].
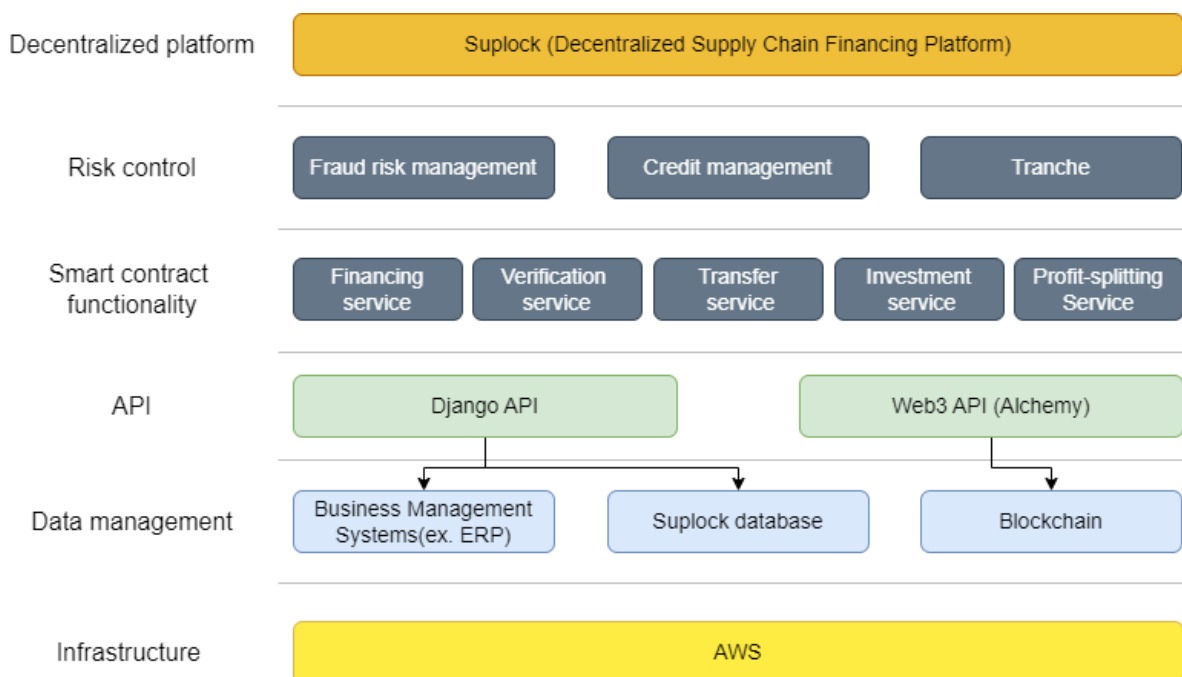
Figure 9. System structure.

5.2. Blockchain and Smart Contracts

Blockchain is a peer-to-peer decentralized mechanism [9], making it very suitable for the internet community (P2P, person-to-person) network platforms, or in other words many-to-many social network platform. There is no universal definition of smart contract, in general, a smart contract is an agreements or a transaction protocol which can be written by computer program to automatically execute, control or document events and actions according to the terms of a contract or an agreement [70].

In this study, four smart contracts are deployed for the account of currency token (introduced in Definition 1), the operations of CT, verification and for the investors.

Currency token (XT): this smart contract deals with the cryptocurrency XT issued by our system. All of the transactions submitted through our system are paid in XT. (Deployed at 0x601cdEcd235a43F9181A8213B2a664A9703a742e).

CT operations: this smart contract serves as an oracle to access data from business management systems and mints obligatory-type CTs whenever finance requests or transfer requests are submitted. Each core manufacturer(guarantor) has to deploy its own smart contract, e.g., Core Company Core1's smart contract deployed at 0xEE02f04A9b5c500dC915D54e1Fce79d016795526.

Verification: this smart contract serves as an oracle to access data from business management systems and mints obligatory-type CTs whenever a verification of inventory is passed. (Deployed at 0xf2b706d5e3c828D592Db44980Fbd8899D6490938).

Investment and profit distribution: this smart contract deals with the funding certificates. After a funding provider provides funds by XT, this contract will mint the corresponding right-type CTs to funding provider; and once the system receives repayment from the financing company, this contract will share the profit based on right-type CTs. (Deployed at 0xf788E3cEdfc6E66Cd553CE9636d418F141992DB2).

5.3. Financing Workflows

SCF system is operated by the means specified by business management systems and the use of public chain. The business management system is appointed to serve as the means to provide data that stems from the supply chain and also act as the source to provide verification. While the blockchain is used as a solution to provide traceability to the financing events to mitigate risk events. Based on the data flow combined with the

consensus algorithms, we are able to provide below mentioned mechanisms for supply chain financing on the public chain.

Channel financing: The core company will mint obligatory-type CT with specified conditions on the blockchain to provide guarantee and transparency, then the receiving downstream company can use it for further finance or transfer. The workflow is shown in Figure 10.
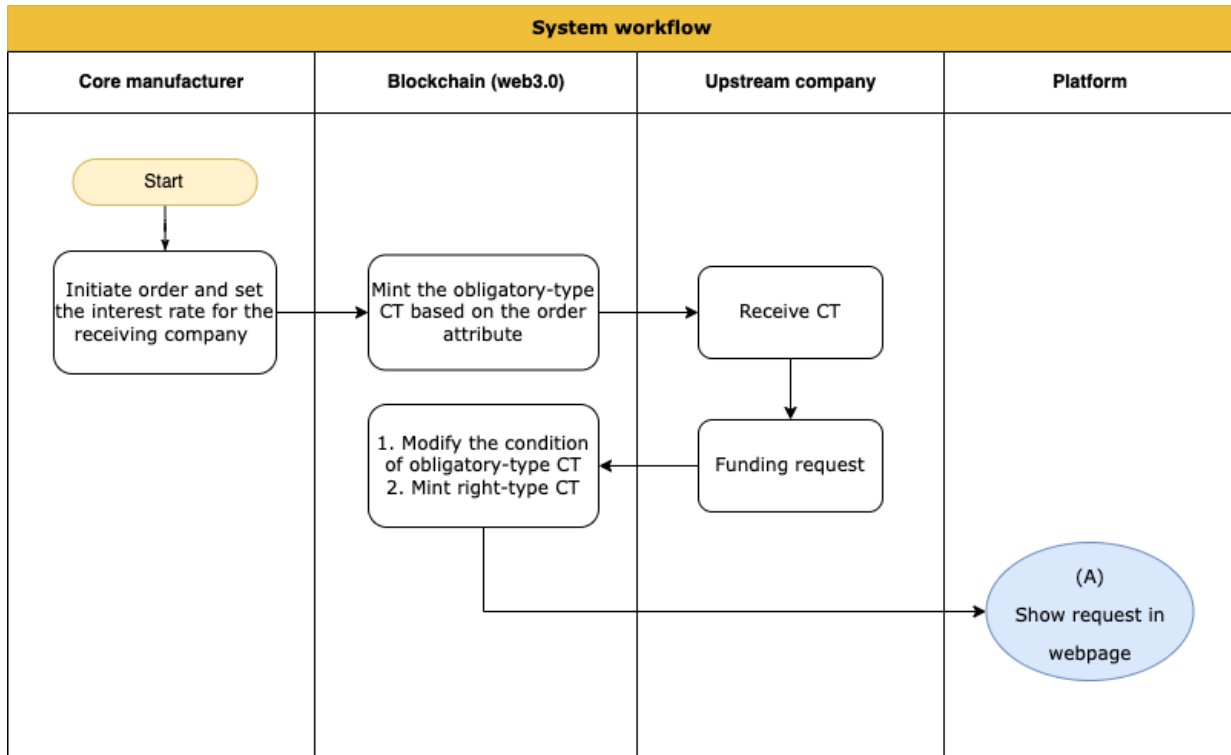
Figure 10. The workflow of channel financing, where (A) is the results shown in the webpage, and is used to list all of the funding requests. So that the funding providers can browse this list and connect to the workflow of funding as in Figure 11.

Accounts receivable financing: The obligatory-type CTs are minted to reflect the time value, therefore discounts are applied in the financing process.

Inventory financing: The obligatory-type CTs are authorized by a third-party institution. The inventory is used as the collateral which will be liquidated after a breach of contract, then the funding provider(s) would be able to retrieve payback from the inventory liquidation amount.

Investment service: Natural person or juridical person can be secured via right-type CTs after providing funds. Every right-type CT offers different annualized payment yield (APY) by investing in different tranches for different ranking of claims. The workflow is shown in Figure 11.

Transfer and conversion of CTs: CTs can be transferred via blockchain as the debt transferring. Companies will be able to manage their obligatory-type CT received from the previous company to conduct risk-control more easily, meanwhile to maximize the efficiency of the supply chain. The workflow is shown in Figure 12.
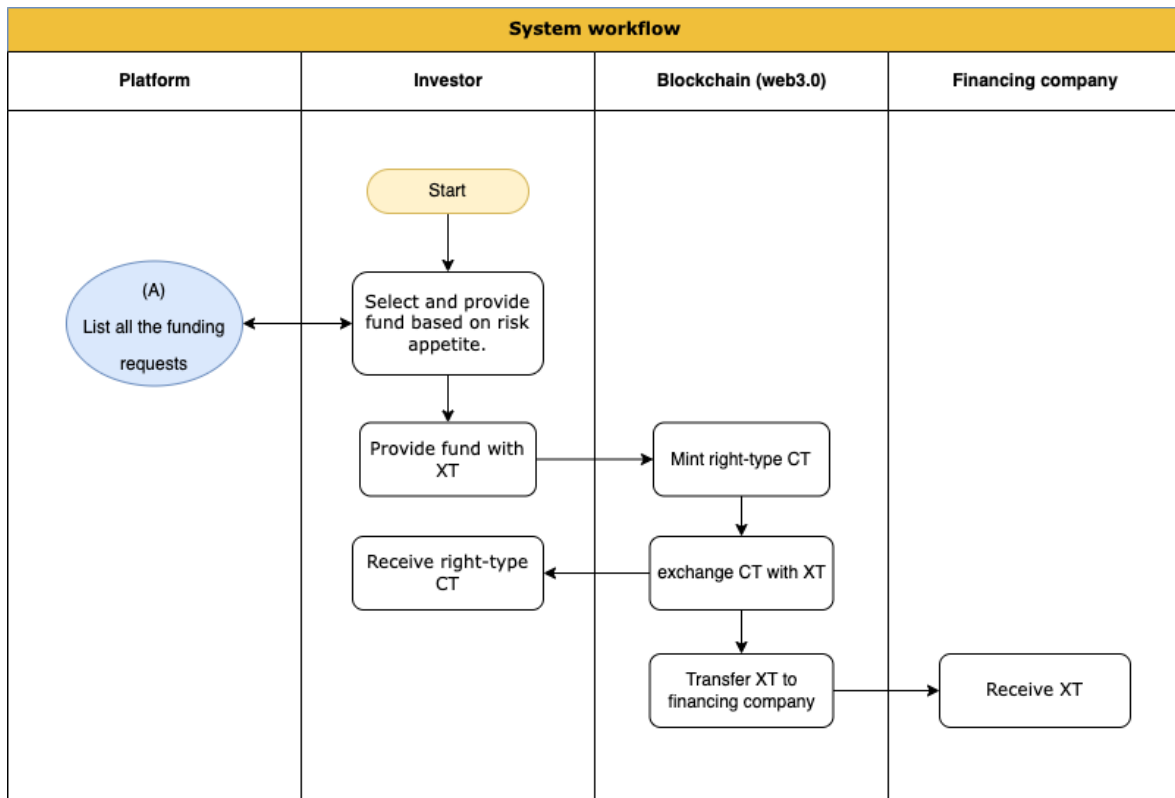
Figure 11. The workflow of providing fund, where (A) is the list of funding requests described in Figure 10.
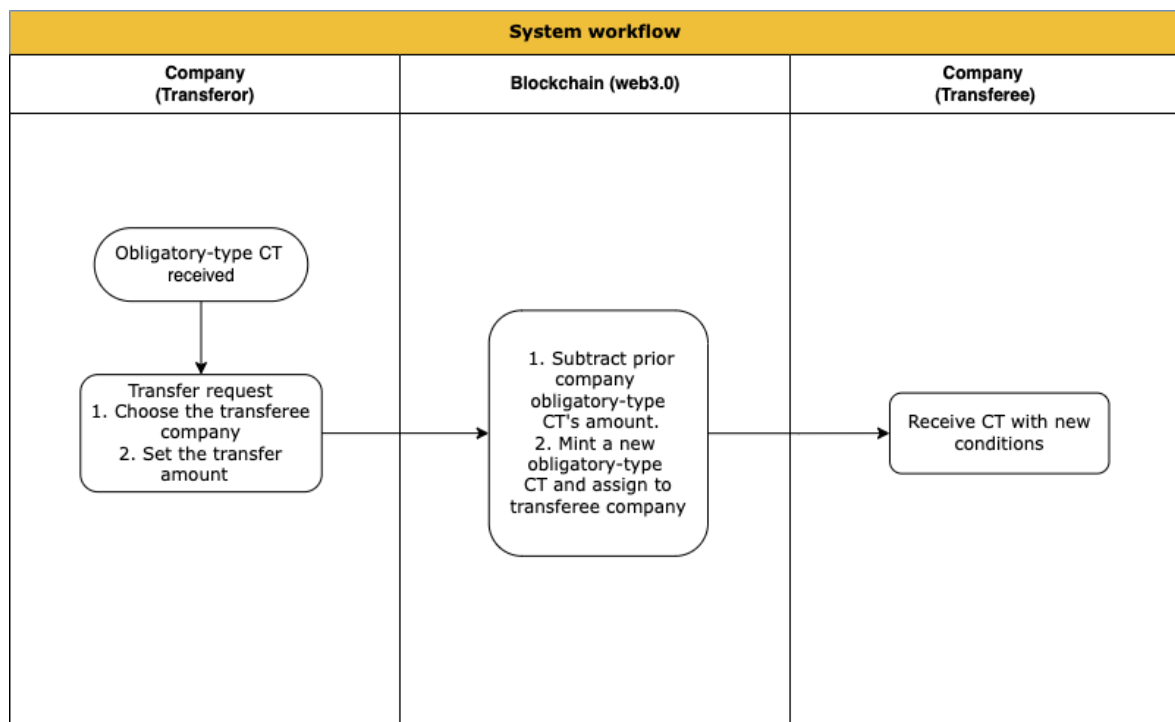
Figure 12. The workflow of transferring and the conversion of CTs.

Finance management: Profit sharing agreement between core company and investors is programmed in smart contract. The workflows are shown in Figures 13 and 14.
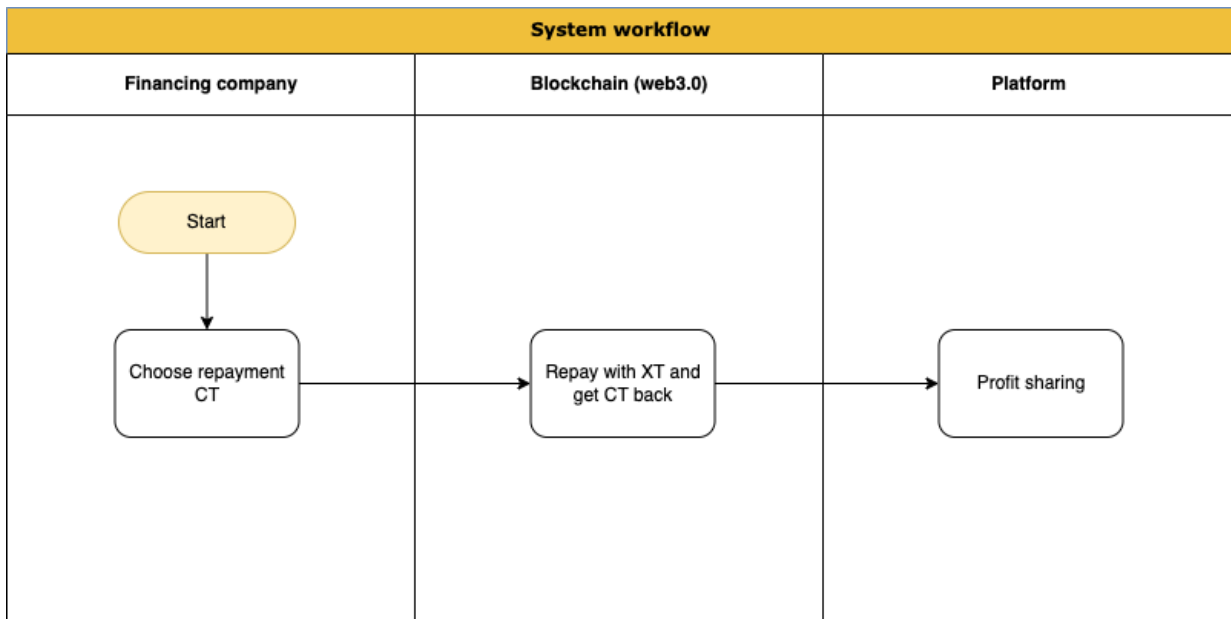
Figure 13. The workflow of repayment in finance management.
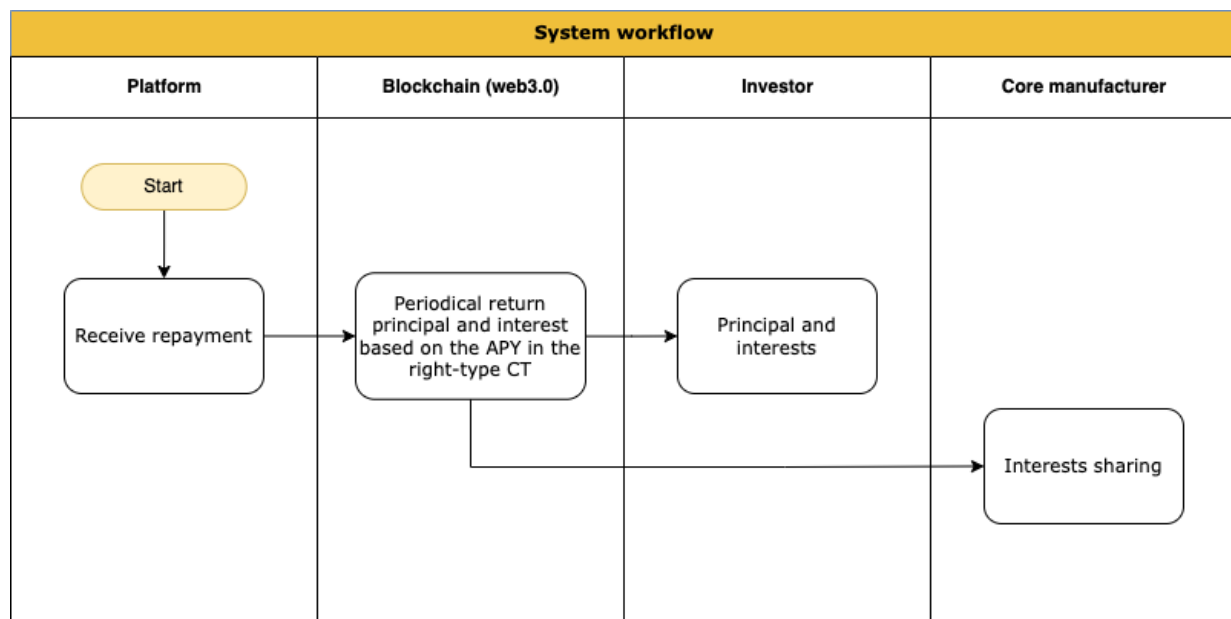
Figure 14. The workflow for profit-sharing in finance management.

Finally, the risk control module is used in case of default, including cases where the user delays repayment, or where the issuer refuses to take responsibility. For the first type the delayed interest and a default penalty charge needs to be calculated as an offset for the value of the goods, while the second type requires auctioning. Both are beyond the scope of this article.

6. Feasibility Analysis

In this section, the legal properties, risk management, and costs and benefits are discussed.

6.1. Legal Analysis

CT used to represent the legal right called "verdinglichung obligatorischer rechte" will be discussed below.

6.1.1. Rights in Rem

As participants in supply chain financing core companies act as someone providing collateral or guarantor, providing the collateral to set up the mortgage, for the CT to be issued in accordance with the mortgage, providing debt guarantee and bearing the final responsibility of full repayment. Among them, the collaterals can be any tangible or intangible asset.

In the inventory CT or collateral CT, the collateral is created by smart contracts and recorded in the blockchain to fulfill the requirements for rights in rem. In addition, the data on the blockchain cannot be tampered with or forged and can be traced back to its source, so that rights in rem can be installed in a similar way of installing rights in rem in the real world. In practice, in response to different legal requirements in various countries for the rights in rem, it is also possible to connect to the authority officially appointed by the government through using the application interface for the rights in rem. The blockchain can provide a proof of existence of these rights.

A line of credit mortgage is a mortgage created for a specified maximum amount by using a debtor's or third party's assets as guarantee and thereby secure a creditor's rights against the debtor. Therefore, there are three elements: one is the assets used as collateral, second is a specific maximum amount, the third is unknown creditor's rights. That is, the debtor can borrow unspecified amounts of capital as long as it remains within the specified maximum limit and bear the responsibilities (interest and repayment) as agreed upon in the loan terms.

The number and specific conditions of the CT are recorded on the blockchain. When the CT is used, the agreed upon conditions are fulfilled, meaning that the use of the conditional token requires the fulfillment (exercise) of the obligations (rights) endorsed by the set conditions at the same time. An obligation-type CT, providing an (margin, inventory, etc.) as a collateral, setting the number of tokens that can be issued (that is, the maximum limit), a liability only when using (unspecified), plus the public nature the public chain, is like a true line of credit mortgage. In addition, CT's are issued in multiple quantities and can be transferred to third parties. Each CT represents a dividable (securitized) line of credit mortgage. While the right-type of CT can act as a divisible fixed income bond.

6.1.2. Obligations

Clearly, the obligation-type and right-type CT's will install a true creditor-debtor relationship as long as the right conditions are set and fulfilled.

Traditionally, disputes between claimants and debtors, needed to be resolved through judicial means. However, with the use of CT, the smart contract will force the creditor-debtor relationship to be activated, with the smart contract sending the transaction to the blockchain, where the debt is recorded. Keeping track of creditor's rights and debts in public, is equivalent to create of creditor's rights in rem and can resolve disputes between two parties easier.

6.2. Risk Analysis and Management

The need to fulfill conditions set for CT's comes from a user's default or fraud risk. The source of default comes from the guarantor bearing the ultimate responsibility. While fraud mostly happens in case of having double collateral and mortgages.

Investment CT is used for funding the obligation-type CT. So, the ultimate risk of loss is caused by both the risk of default and from an investment loss.

6.2.1. Up/Downstream

Because the CT is divisible and liability only occurs after use, the manufacturer can decide on how to use CT's according to their specific financial situation: they can decide to not use, or transfer or use all or part of it.

It is even possible to convert the conditional tokens to reduce liability, meaning, modifying its conditions. If the required liability is lower than originally set in its conditions, it is equivalent to a partially using that condition. This will provide flexibility in the use of funds by those who need it. Modifying does not expand the maximum credit, and since the use and transfer of information is recorded in the blockchain and is therefore undeniably true, so as to restrain self-discipline and serve as a risk management mechanism for upstream and downstream manufacturers.

6.2.2. Guarantors

If the CT defaults, the guarantor is ultimately the responsible organization of the conditional certificate. Therefore, the conditional token provides the conditions to use and transfer for the issuer, including but not limited to: interest rate, term or transfer level restrictions, etc. All ensuring that the manufacturer can manage its risk through setting the conditions.

The CT cannot predict a breach of contract between upstream and downstream manufacturers. However, it can provide transparent information on the breach of contract, including the situation on the use or transfer of tokens by the upstream and downstream manufacturers, and thus ensure that the guarantor can use the goods for price reduction after the goods arrive to the warehouse.

6.2.3. Fund Providers

Fund providers undertake the ultimate risk of loss. At this time, risk management can be divided into collateral and investment risk management. Collateral risk management refers to verifying and evaluating the collateral (including credit) when issuing CT's. If market trading exists, the evaluation will be based on the market value. If not, a verification process done by qualified persons or institution is required.

Holders of investment CT can additionally govern and predict through the use of blockchain. For blockchain governance a voting mechanism can be written in a smart contract, with the voting rights being determined by the number of investment CT's held, and the value of the collateral for the issuer being determined through vote. Blockchain prediction uses smart contracts to obtain external information. After the external third party has verified and evaluated the core enterprise, the value of the collateral is estimated and evaluated on its value as a security by the blockchain prediction. Both are unique methods of the blockchain, and can be used as methods for value assessment.

Fund providing to SCF is similar to investing in fix-income securities, in both an investment risk exists. In this article, we look at the use of tranche investments and pooled funds as a way of managing investment risk. At least three types of fixed-income securities exist, minimum risk (tranche A), mezzanine risk (tranche B) and residual risk (tranche C). If funds are early redeemed or defaulted, then the principal of tranche C will be returned or lost first, consequently followed by the principals for tranche B and tranche A. The risk linked to tranche A is the lowest, and for tranche C the highest. Therefore, depending on how much risk a fund provider wants to take for what risk reward, a different undertaking of each tranche to manage their risk.

Pooled funds are another investment risk management tool for fund providers. Because there are different types of obligation CT's in the pool, the risks for fund providers investing in the pool are being reduced due to diversifying the funds into different CT's.

In addition, the owner of the CT can also transfer the investment CT's to others, and with that transfer and disperse his risk.

6.2.4. Fraud Risk Management

Another risk management advantage of using CT is that cases of double collateral or guarantees can be avoided. The smart contract can issue a token which needs to be linked to the goods that are provided as collateral by the guaranteeing enterprise (called a goods token) and is used like a barcode. Because the records of the blockchain cannot be tampered with or forged, providing the goods as collateral is equivalent to executing the lock-in function of the smart contract, and thus sending the transaction on the blockchain. Now, the double collateral is equivalent to sending a transaction of the goods token twice, it is impossible to execute because there is no specific token for the second transaction. The use of CT's means that the supplier and the issuer generate claims and debts. Similarly to the explanation for goods tokens, a smart contract cannot execute the same guarantee twice.

Finally, inventory CT's and collateral CT's are recorded on the blockchain, therefore they can be used as a basis for credit scoring and risk management.

6.3. Costs and Benefits

The cost of the conditional token is divided in costs for deploying smart contracts and transaction costs in the public blockchain. With the latter being the most important costs, necessary to change the state of the blockchain (aka the change of the ledger).

The costs of deployment are determined by the amount of code in the smart contract and is a one-time cost. The calculation method comes down to multiplying required gas units with the gas price per unit of it. Take the most popular Ethereum chain (Ethereum) as an example, the total gas fee is equal to the basic gas limit of 32,000 units plus the gas (base) fee of 200 units per byte [71]. The gas price is divided into three categories: fast, normal, and slow referring to the required deployment speed. The recent average price from 1 November to 30 November 2022 for gas is 23 gwei [72] (1 gwei = 0.000000001 ETH), making the market price around USD 0.000029 [73] (1ETH = USD 1276).

In this study, the smart contract of CT is deployed on the Ethereum test chain, the deployed address: 0xEE02f04A9b5c500dC915D54e1Fce79d016795526. It's gas fee is about 4.2 million refer to Figure A2 in Appendix A, or 0.096 ETH ($= 4.2M \times 23$) which equivalent to USD 123. However, these costs can be ignored for each transaction after amortizing to the number years of financing.

The transaction fee of the public blockchain depends on the type of chain and the number of transactions. Taking the Ethereum chain for example, between November and December of 2022, the fee for each transaction was 0.0014–0.0023 ETH or USD 1.7–2.82 (Figure 15). At the same time, the collateral fee rate for Taiwan Small and Medium Enterprise's is 0.375% [74], thus the fee is USD 37.5 if borrowing USD 10,000. Comparing with the handling and international trading fee of the financial industry, the CT fees are supposed to be lower for the following reasons:

1. There are variety of costs for implementation and running business in traditional SCF [75], only the fee charged by platform, including transaction fee on blockchain in CT.
2. The transaction costs are high for traditional SCF. The report of International Chamber of Commerce (2020) [76] shows 83% of banks concern high transaction costs or low fee income on trade finance in future.
3. The losses and the number of people caused by fraud should not be underestimated. For example, the case Yasin in Taiwan has more than 6000 people who filed a lawsuit (98-Jin-3, Taiwan Shilin District Court 2015). If we consider the confidence from general public in SCF, it is even more impossible to compare.
4. Many studies [11,18,77] show the cost effective if apply blockchain technology to supply chain finance, especial for deep-tier companies.
5. CT can improve capital efficiency in society. Traditional SCF seldom provides funds for manufacturers above the second tier or deep-tiers due to risk considerations. However, by the transfer operation, CT can help to increase the sources of funding

for deep-tiers and let non-financial entities participate in the role of fund providers, overall improving economic efficiency. The cost of funding thus will be decreased.

Figure 15. Average daily transaction fee of Ethereum blockchain from 22 November 2022 to 24 December 2022 https://blockchair.com/ethereum/charts/average-transaction-fee-usd (accessed on 15 January 2023).

Concluding the above, we might claim the benefits of using CT will reduce losses and improving supply chain efficiency, even there is no global statistics on costs of SCF.

6.4. Comparison

The biggest difference between the public blockchain and the consortium chain is its degree of inclusivity and credibility. The source of funds of the consortium chain comes mainly from the financial industry participating in the specific consortium chain. Additionally, because of contradicting with the decentralized nature of blockchain, and the financial industry's relationship with credit rating the consortium chain cannot be extended towards the public. Fund users also need to be members of the consortium chain. However, CT can be extended to the public and to any of the vertical or horizontal supply chain manufacturers, with that achieving the goal of true inclusive finance.

In terms of credibility, the source of credibility comes from the number of nodes in the blockchain, and the nodes of a consortium chain are far less than the nodes of a public blockchain, so data is more likely to be controlled by more than 51% of computing power, indirectly affecting inclusivity.

In addition to the above-mentioned advantages of the public chain, the CT also increases flexibility because of its transferability, amenability, and the required conditions to be fulfilled to undertake obligations or to enjoy rights.

Summarize the above in Table 3.

Table 3. Methodology comparison.

Items	Traditional	Consortium Tokens	Conditional Tokens
Tier	1 + 1 + 1	N + 1 + N	N + 1 + N
Safety (consensus)	centralized database	asymmetric cryptography	asymmetric cryptography
Cash flow transparency	tier 1	N + 1 + N	N + 1 + N
Credibility	single financial institution	permissioned member	unspecified person
Flexibility	law	middle	high
Inclusive	law	middle	high

7. Concluding Remarks

The supply chain provides logistics and flow of information, and includes the vertical supply chain N + 1 + N and the horizontal supply chain of international trade. However, due to the difficulty of credit investigation, the financial industry is limited to SCF focusing on core enterprises in terms of capital flow support.

This study proposes a new model for SCF based on CT issued by smart contracts deployed at the public blockchain. The study explains a method on how to use CT's, how risks can be managed, and implications for legal compliance, system construction and cost-effectiveness. It is believed that CT's provide (1) credibility, by adopting the public chain resulting in the highest degree of credibility. The public nature of the blockchain can be exploited to guarantee the rights of securities, mortgages and even creditor's rights. (2) Flexibility, to set or reset conditions of use and adjusting amounts for transfer or use to meet desired risk requirements, effectively solving problems of financial institutions in financing throughout vertical and horizontal relationships. (3) Security, blockchain use asymmetric public-private key cryptography, therefore CT's can be used as security verification mechanisms. (4) The transparency of cash flow, the source, destination and authenticity of CT's can be confirmed through the blockchain, effectively preventing money laundering and suppressing fake transactions.

Based on the above statements, the application of CTs to internet-based community (P2P) or e-commerce financing platforms will effectively remove barriers in the application of SCF. It is applicable to non-specific fund demanders and providers, and it is universal and marketable. In addition, CT's can of course also be applied for consumer finance, or crowdfunding. In reality, smart contracts for consumer finance (deployed address: 0xf6daae7777472be83633329c2733a9246ab6a1b1 for currency tokens, 0x8c51c69a6bc5d4836da8433ceefdc80e7e60ddd7 for obligation-type CTs, and 0x45176c52685298ae9e6944bfe67bc3924f12e1bb for obligation-type CTs, introduction video is at: https://youtu.be/3O9Q8kwgsAo, accessed on 21 February 2023) and SCF (refer to Appendix A) have been deployed in the test net, as well as on concept of proof platforms.

In addition, once the CT market is large enough, then CT's can also be used in market transactions, just like crypto currencies generally do. The CT's nature is much like a manufacturing issuing corporate bonds, with the market price depending on the financial status of core companies and on market interest rates. It will lead to a maximum capital efficiency in SCF, requiring further development in the future.

8. Patent

Taiwan patent M607838 Financial service system, and M602237 Matching-Service System and Account Handling Device.

Conflicts of Interest: The authors declare no conflict of interest.

Appendix A. Illustration Example of SCF Test Platform

In our examples we use four types of smart contracts. The first type will be deployed to issue exchange currency token for users. The second type will be deployed for the core company which is used to issue and operate the obligation-type CT. The third smart contract is a kind of ERC1155, which will be used to validate and issue the collateral. The third type is used to issue the right-type CT. All of the smart contracts are deployed on the Goerli test net.

Appendix A.1. Core Company Core1

Appendix A.1.1. Wallet Address (Aka Public Key)

0xf0d0F12f561e879c1c3098fFe2762Fa1AC59d901.

Appendix A.1.2. Deploy Smart Contract

Core1 requests 0x8144749b02bd30885b11c20789f115d58ffc0aed (aka platform) to deploy smart contract, platform deployed it at block 7783518, at the address 0xee02f04a9b5c500 dc915d54e1fce79d016795526, or linked at https://goerli.etherscan.io/tx/0x26f75067d11f8 c90d424855555d67a536a411b65e72e7fe83a3a88171857c60a (accessed on 15 January 2023), for which the transaction fee is 0.000425011523502264 Ether. The transaction details are as shown in the Figures A1–A3.
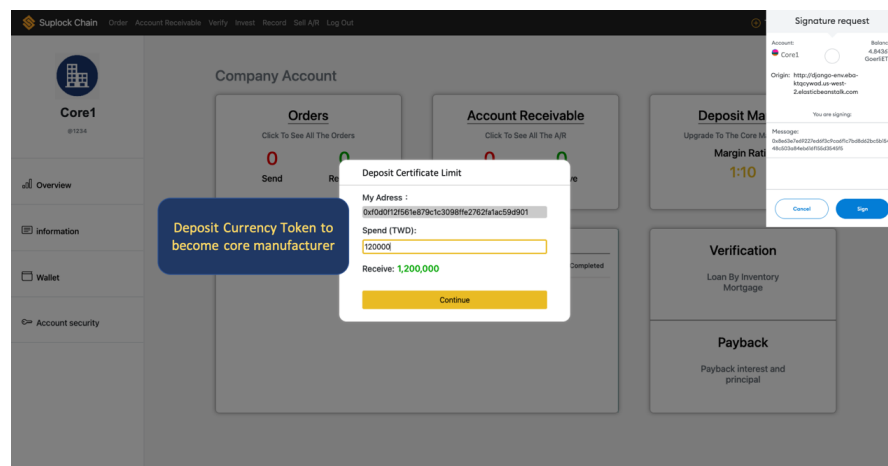
Figure A1. Module for company depositing credit and deploying smart contracts. Core1 requests to deploy smart contract on blockchain.
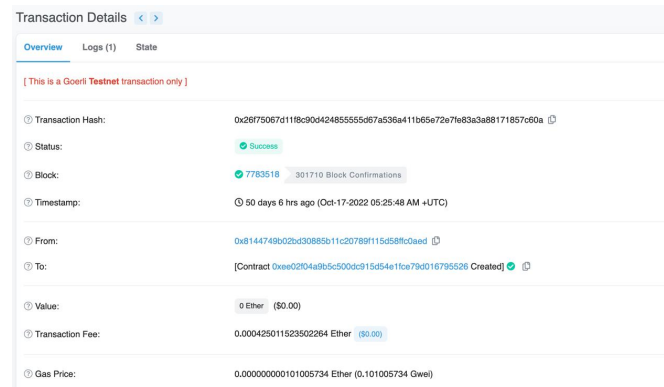
Figure A2. Proof of deploying smart contracts on Etherscan.
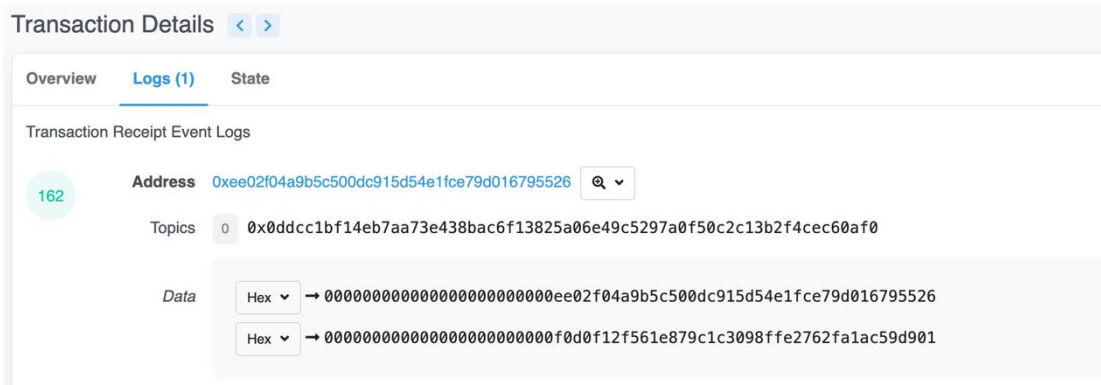
Figure A3. Logs of contract creation transaction on Etherscan.

Appendix A.1.3. Mint Currency Tokens

Currency tokens are defined in "Definition 1". The platform, on behalf of Core1, sends transaction to mint currency token 1000 at transaction block 7783559, or transaction hash 0x5990db2d0c6aff04a13a8da7f06c85ee8f0e5ff951caeb5eeb06e23c5b4998a3. An overview of the information is given in Figures A4–A6.
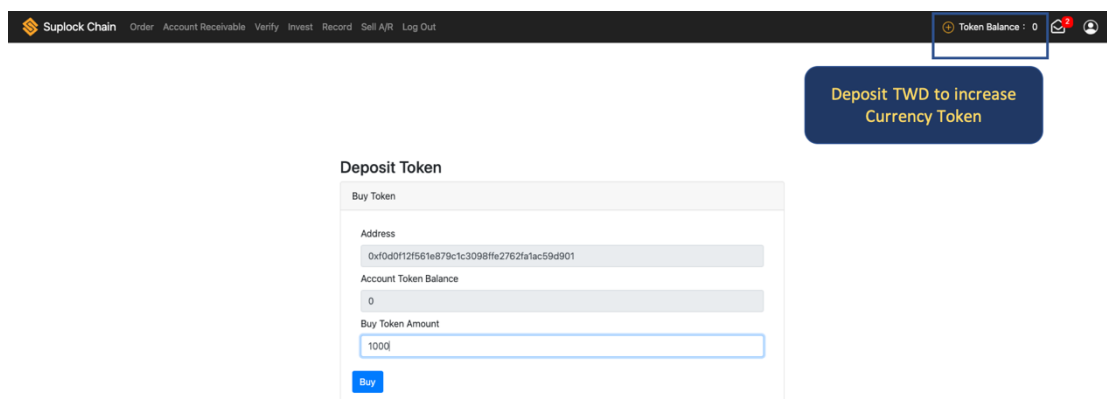
Figure A4. Module for minting currency tokens. Core 1 requests to mint currency token on our website.
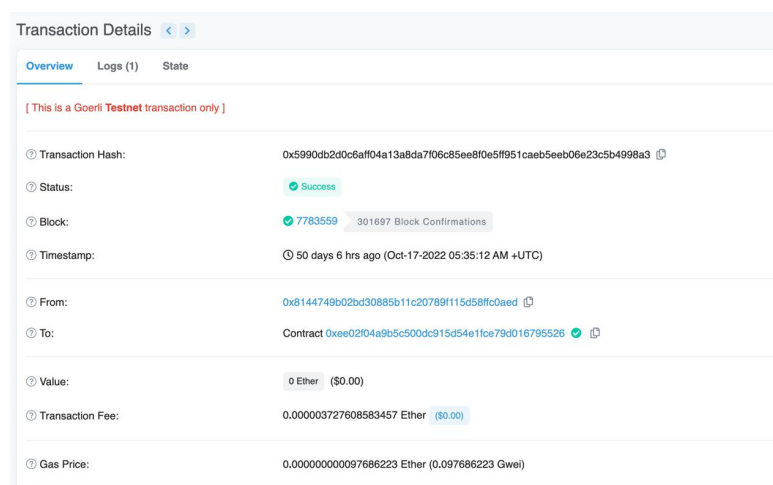
Figure A5. Proof of minting currency tokens on Etherscan.
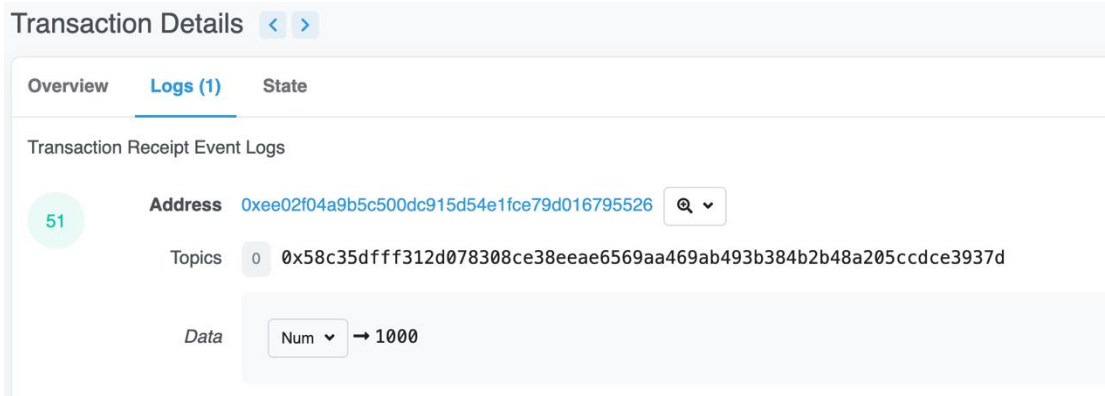
Figure A6. Logs of minting currency tokens transaction on Etherscan, amount was shown in the Data field.

Appendix A.1.4. Mint Conditional Tokens

The minting operation is in Figure A7, the first line in "Data" is the issuer's address (aka core company), the second line is the receiver's address (aka tier 1 company), followed by the transfer amount, class, token id, interest and date. Where class is the type of token, a type 1 is accounts receivable, a type 2 is a purchase order, a type 3 is a transfer from non-core, and a type 4 is used for loans. Below shows Conditional tokens (refer to Definition 1) are transferred to wallet 0x48f0966ad2d0c575863ab84823e1d4fcaa080b51 (a.k.a. Tier1, refer to Appendix A.2).

The platform, on behalf of Core1, send transaction to mint conditional token 1000 at transaction block 8021726, or transaction hash 0xc7f20c39406e2b6c74dbeb372a77e67d281e3e 59d650bc11d821a029ca74e9a5, set up the conditions, and then transfer it to the tier 1. The ledger status is in Figure A8, and the result details are in Figure A9.
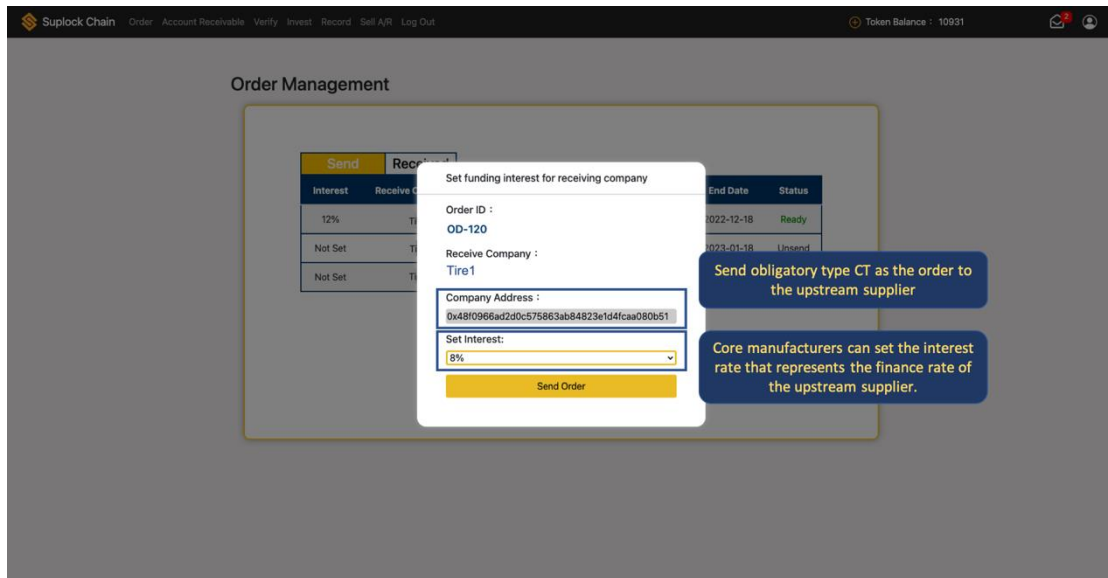
Figure A7. Module for minting conditional tokens. Core 1 issues order transaction to Tire1 on our website.
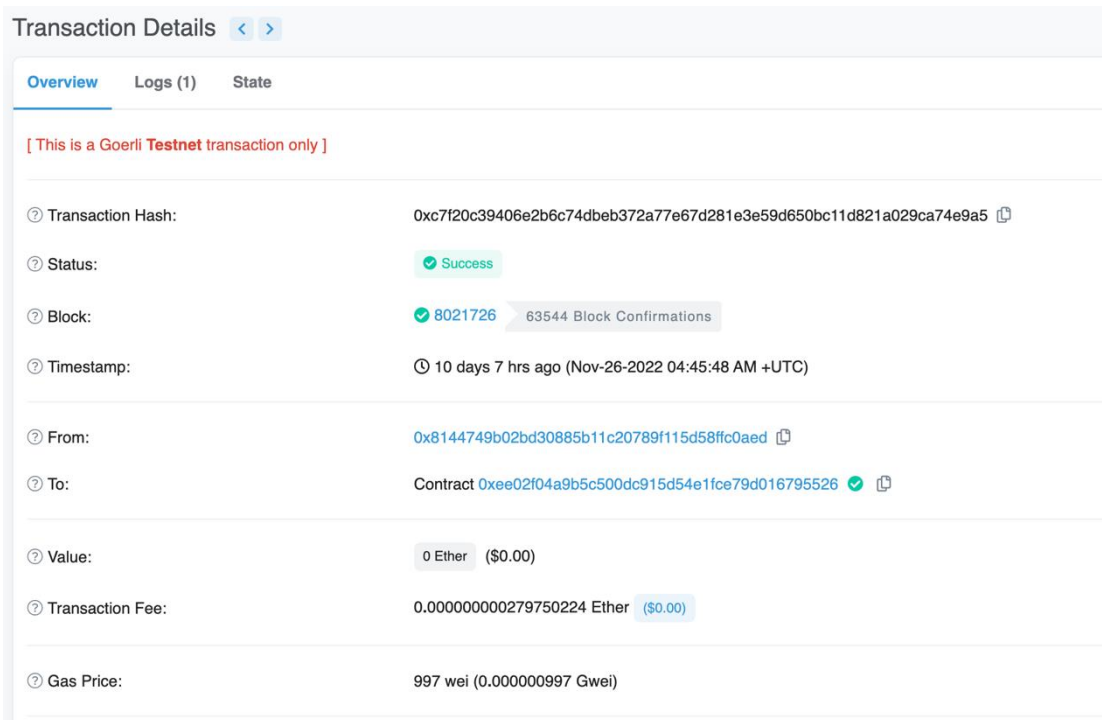
Figure A8. Proof of minting conditional token on Etherscan.
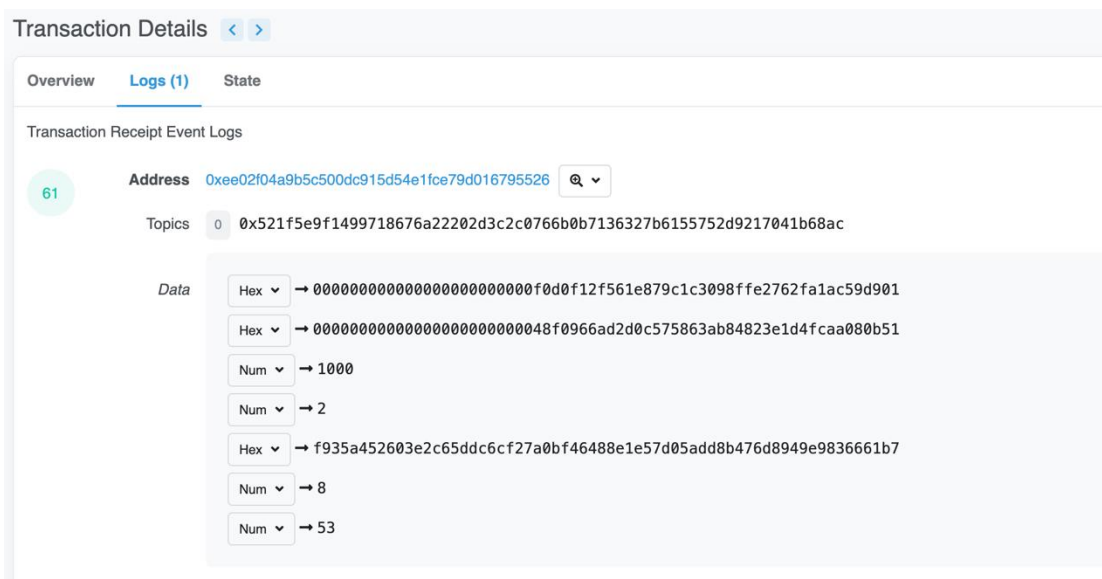
Figure A9. Logs of minting conditional tokens transaction on Etherscan.

Appendix A.1.5. Funding Request

Figure A10 is the operation of funding request (loan request) 1000, Figure A11 shows this request is at block 8021794 from Tier1. In this transaction, there are two processes shown in Figure A12. First, the collateral (current is currency token) needs to be deposited to the platform for custody logged first as "10", followed by setting up the loan id (2nd line), amount (3rd line), interest (4th line), and time stamp under "Data". The log information is in Figure A12.
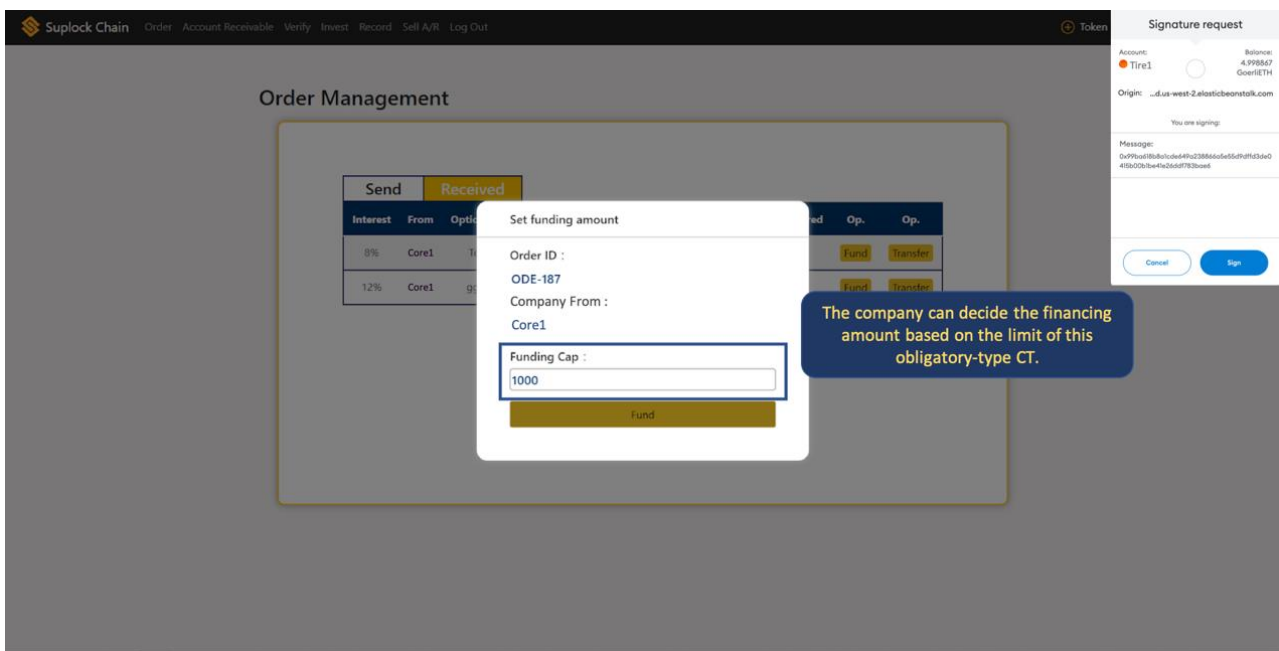
Figure A10. Module for issuing funding request. Tire 1 request funding by collateral order from Core 1 on our website.
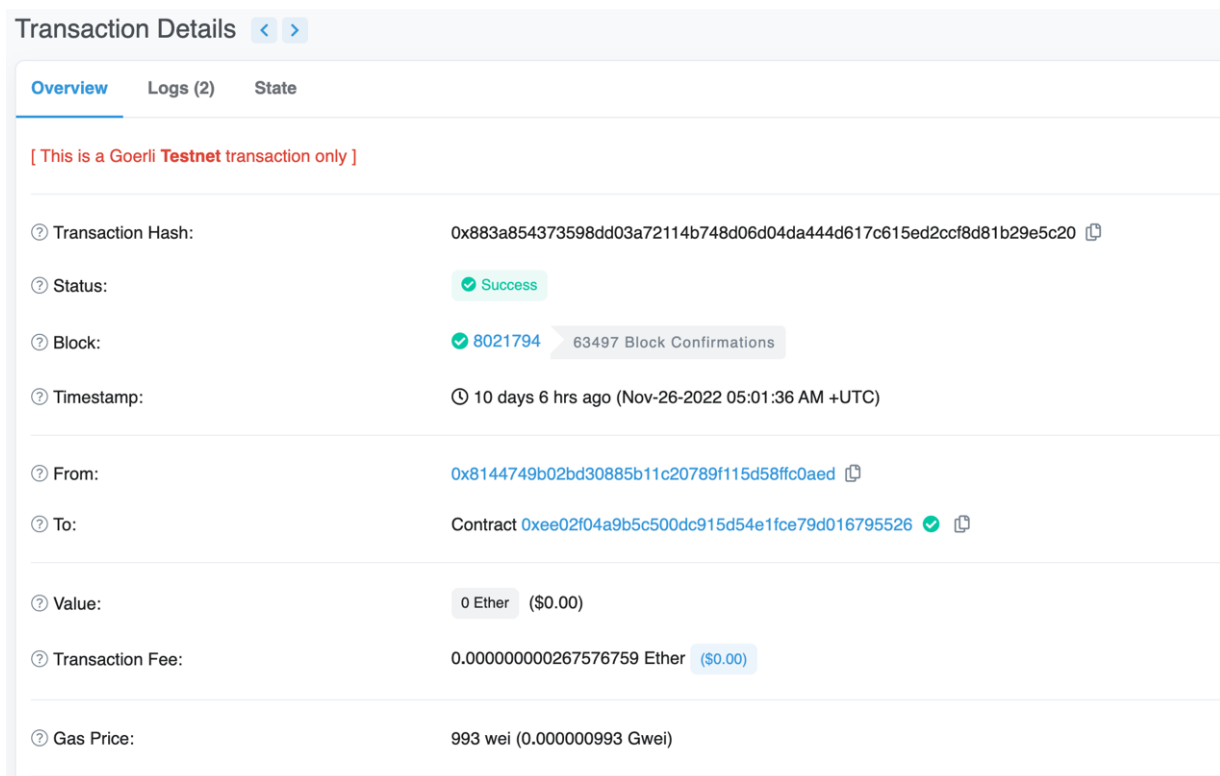
Figure A11. Proof of issuing funding request on Etherscan.
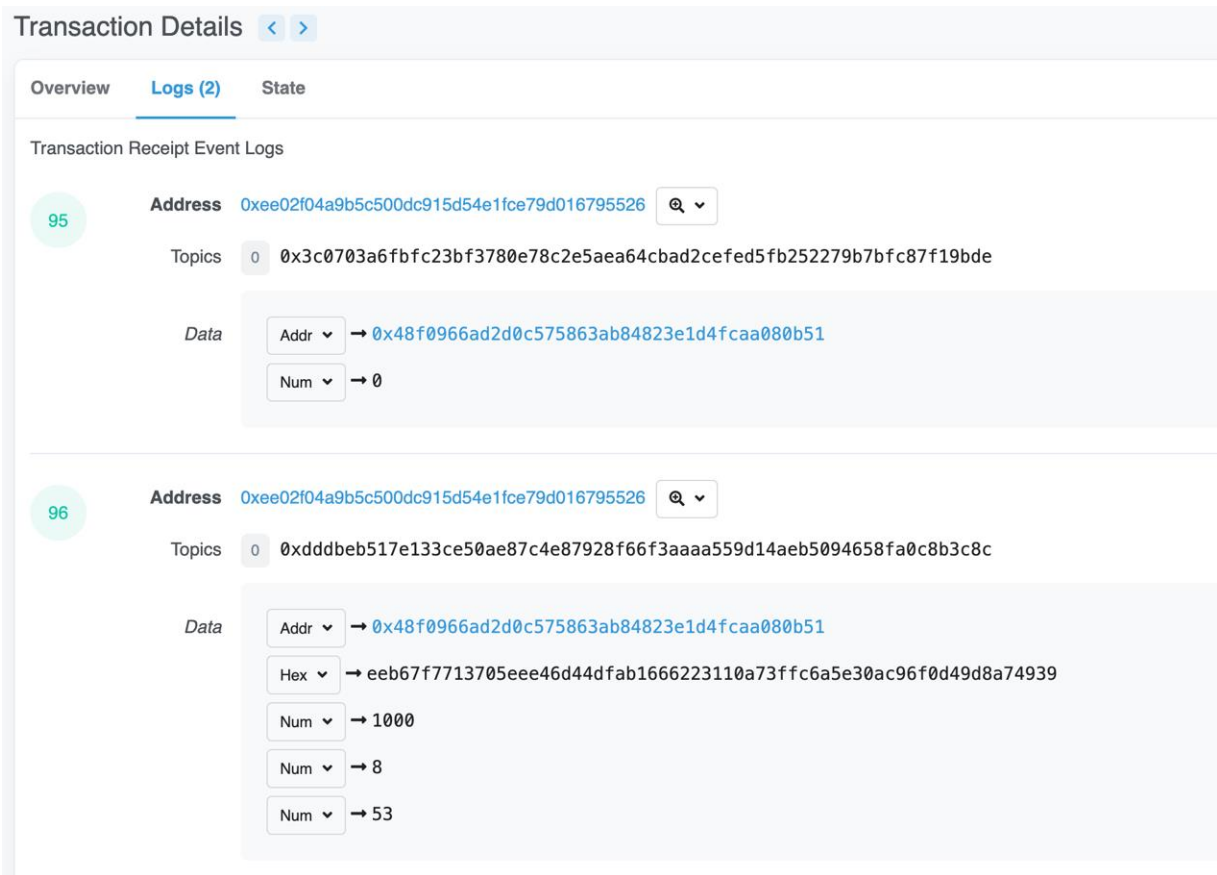
Figure A12. Logs of issuing funding request transaction.

Appendix A.1.6. Transfer CTs from Tier1 to Tier2

Figure A13 is the transfer operation. The wallet of the Tier 2 company is 0xcc 16a9FBE3aBa14E976323E6082f70FA852eC6Ad, and the result is in Figure A14.
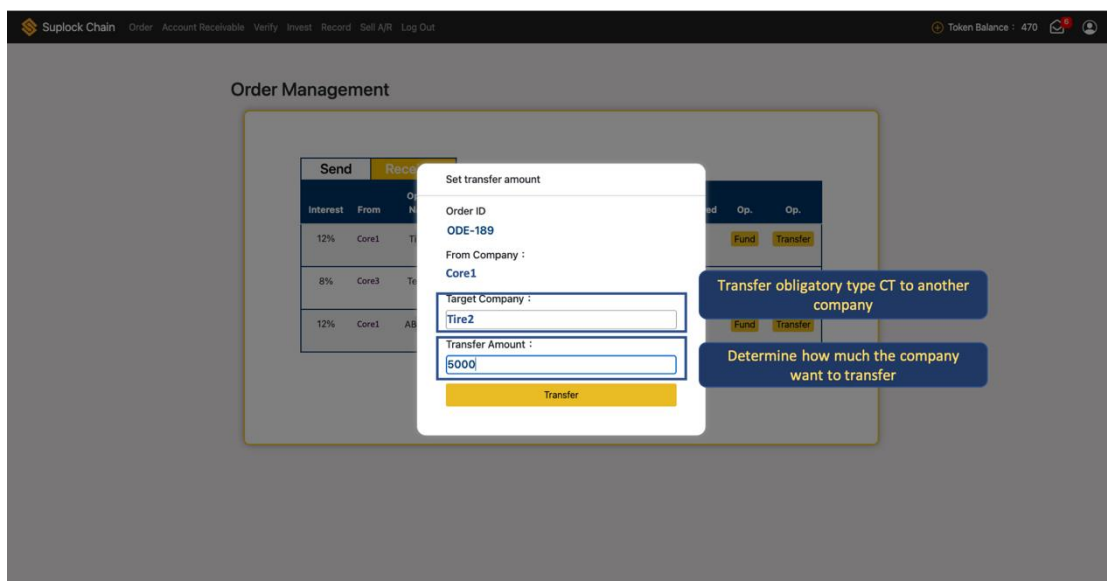
Figure A13. Module for tier1 company transferring CT to tier2 company.
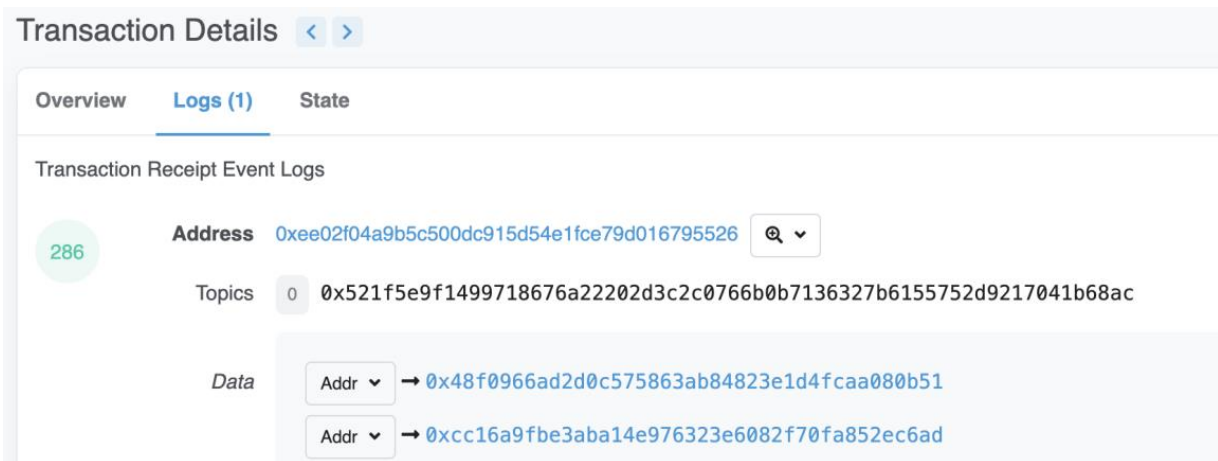
Figure A14. Logs of transferring CT transaction on Etherscan.

Appendix A.2. Collateral

Inventories are used as collateral for loans if required as in Figure 2. It can be tokenized by using the ERC1155 protocol. Figure A15 shows the operation for collateral request.

In Figure A16, the smart contract was created at address: 0xf2b706d5e3c828D592Db 44980Fbd8899D6490938. The collateral tokens are issued to proof "locking in" inventories. The transaction details are provided in the Figure A17.
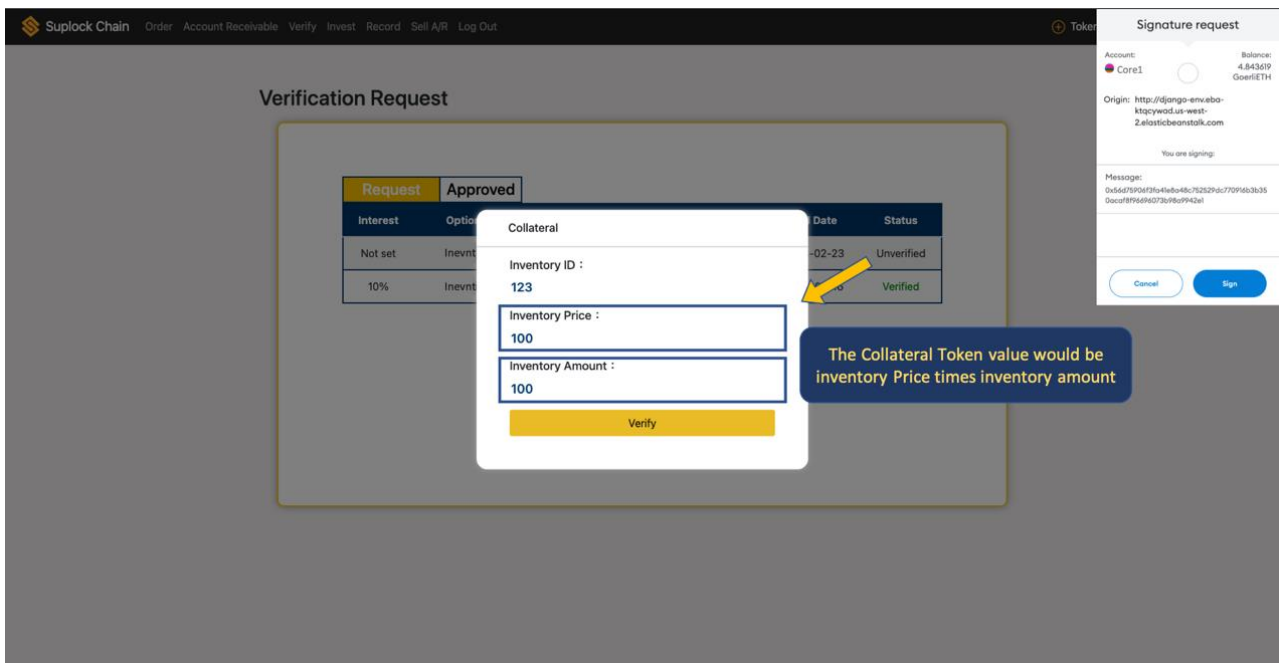
Figure A15. Module for company locking in collateral. Core1 requests funding by collateral inventories on our website.

Figure A16. Proof of locking in collateral on Etherscan.

Figure A17. Logs of locking in collateral transaction on Etherscan.

Appendix A.3. Funding Providers

Funding provider exchanges currency tokens to right-type CTs as in Figure 3.

Figure A18 is the operation menu for funding sponsors. The smart contract address for funding providers is at 0xf788E3cEdfc6E66Cd553CE9636d418F141992DB2, or is linked via https://goerli.etherscan.io/address/0xf788E3cEdfc6E66Cd553CE9636d418F141992DB2 (accessed on 15 January 2023). The transaction information in ledger is in Figure A19.
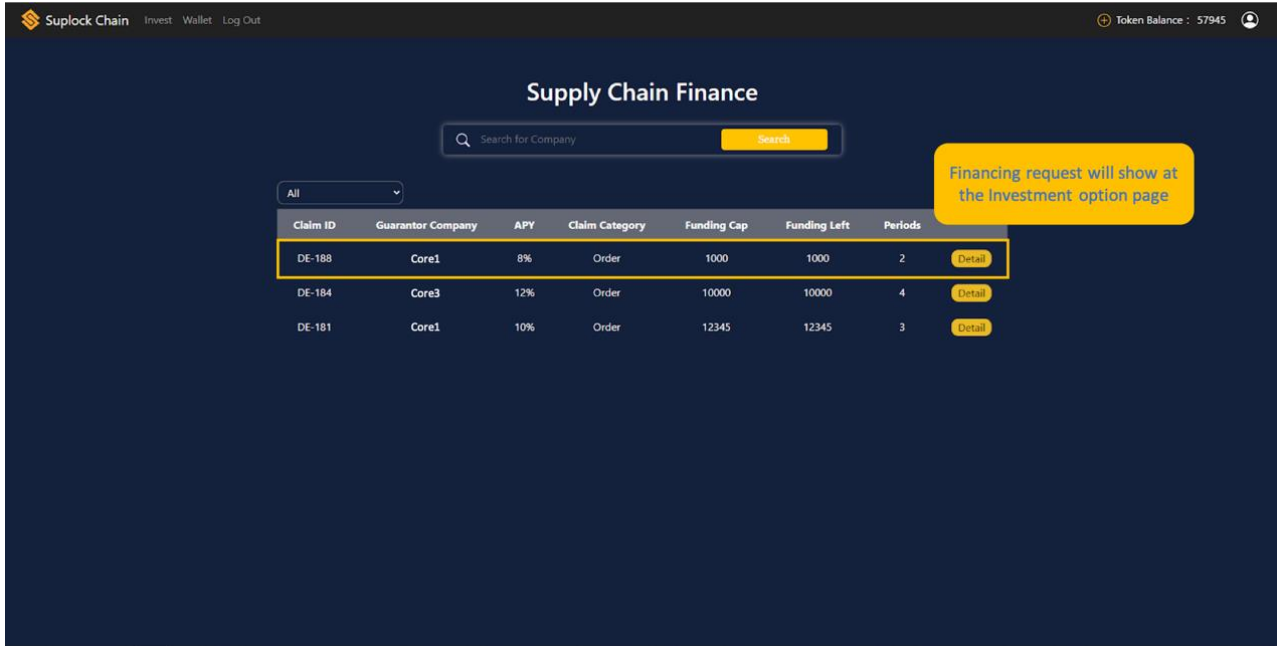
Figure A18. Module for providing fund. Our website lists all the available investments.
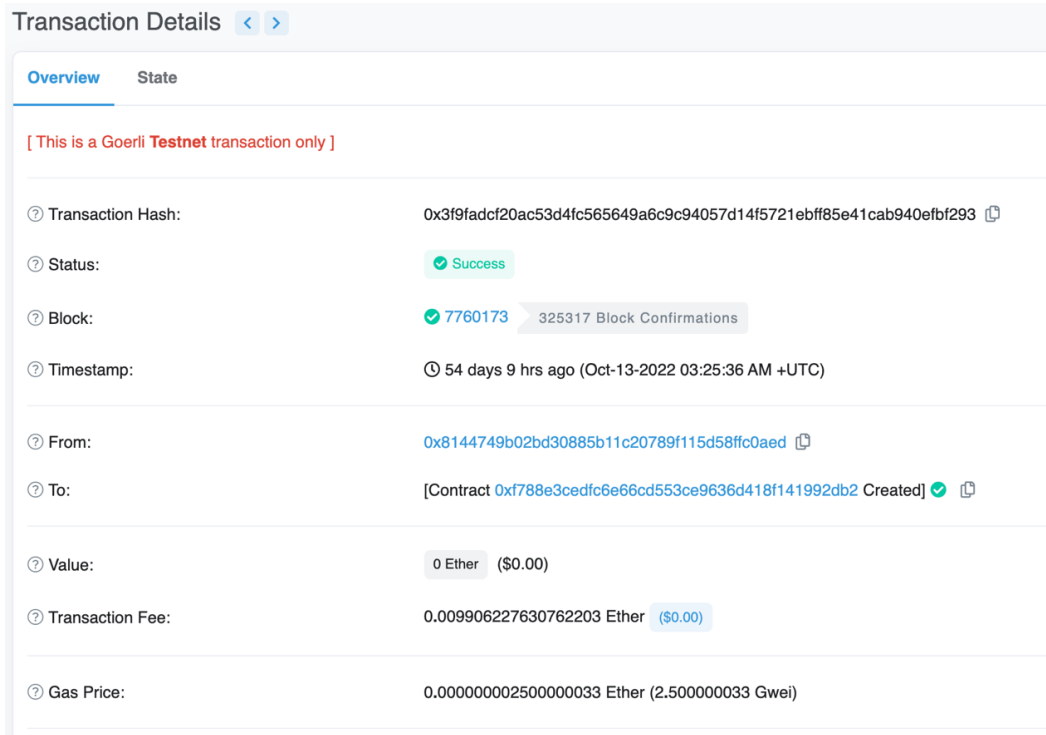

Figure A19. Proof of providing fund on Etherscan.

The available funding requests list in Figure A20, the selected detail is in Figure A21. The funding provider provides fund and receiving repayment are shown at address: 0xb6F835f3d4 FDD8B78179E4d46d1CC61827d60cba, or is linked via https://goerli.etherscan.io/address/ 0xb6F835f3d4FDD8B78179E4d46d1CC61827d60cba#tokentxns (accessed on 15 January 2023). The listed transaction of ledger is provided in Figure A22.
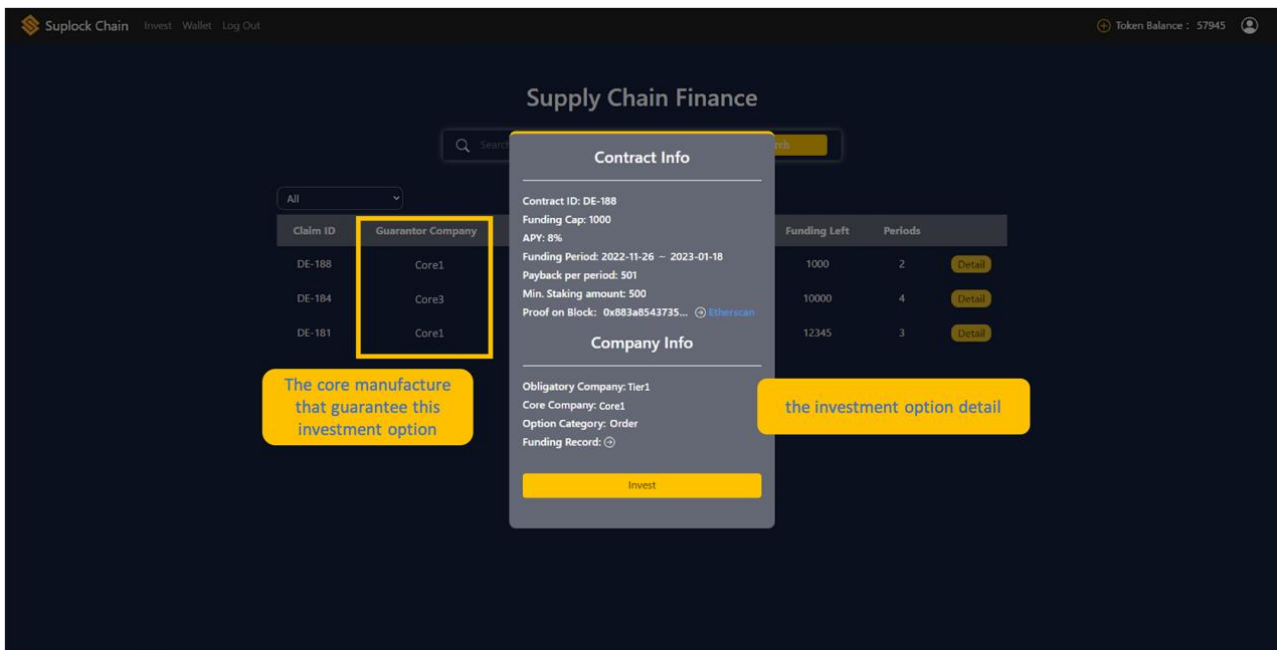

Figure A20. Module for listing right-type CTs that are available for investment on our website.
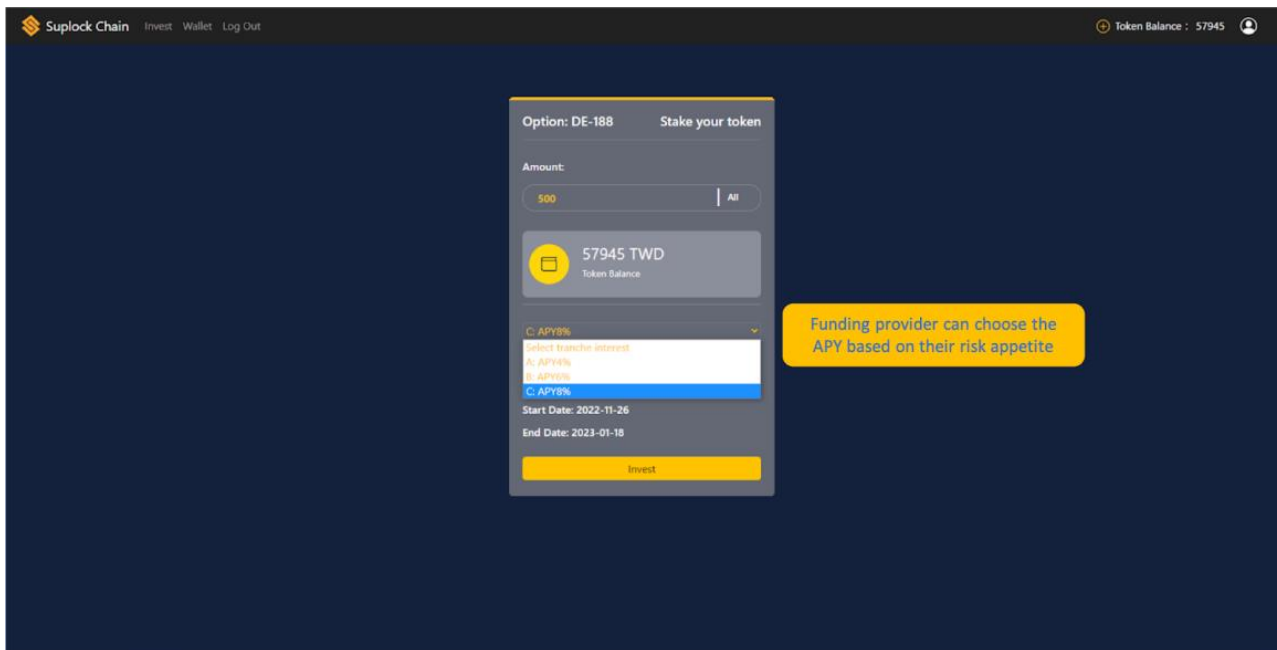

Figure A21. Module for showing right-type CT information on our website.

Figure A22. Proof of providing fund with right-type CTs information on Etherscan.

References

1. Liebl, J.; Hartmann, E.; Feisel, E. Reverse Factoring in the Supply Chain: Objectives, Antecedents and Implementation Barriers. *Int. J. Phys. Distrib. Logist. Manag.* **2016**, *46*, 393–413. [CrossRef]
2. Steeman, M. *The Power of Supply Chain Finance: How Companies Can Apply Collaborative Finance Models in Their Supply Chain to Mitigate Risks and Reduce Costs*; Hogeschool Windesheim: Zwolle, The Netherlands, 2014.
3. Zhao, L.; Huchzermeier, A. *Supply Chain Finance: Integrating Operations and Finance in Global Supply Chains*; Springer International Publishing: Cham, Germany, 2018.
4. Handfield, R.B.; Nichols, E.L., Jr. *Introduction to Supply Chain Management*; Prentice Hall: Upper Saddle Rive, NJ, USA, 1999.
5. Global Supply Chain Finance Forum. 2016. Standard Definitions for Techniques of Supply Chain Finance. Available online: http://supplychainfinanceforum.org/ (accessed on 1 March 2023).
6. Hofmann, E.; Strewe, U.M.; Bosia, N. *Supply Chain Finance and Blockchain Technology: The Case of Reverse Securitisation*; Hofmann, E., Strewe, U.M., Bosia, N., Eds.; Springer International Publishing: Cham, Germany, 2018.
7. Katz, A. Accounts Receivable Securitization. *J. Struct. Financ.* **2011**, *17*, 23–27. [CrossRef]
8. Moller, S. Securitisation—A Safe Bet for Your Assets. *Treasurer* **2000**, 40–42.
9. Nakamoto, S. Bitcoin: A Peer-to-Peer Electronic Cash System. *Decentralized Bus. Rev.* **2008**, 21260.
10. Buterin, V. A Next-Generation Smart Contract and Decentralized Application Platform. *White Pap.* **2014**, *3*, 2–11.
11. Dong, L.; Qiu, Y.; Xu, F. Blockchain-Enabled Deep-Tier Supply Chain Finance. *Manuf. Serv. Oper. Manag.* **2022**. [CrossRef]
12. Choi, T.-M. Supply Chain Financing Using Blockchain: Impacts on Supply Chains Selling Fashionable Products. *Ann. Oper. Res.* **2020**. [CrossRef]
13. Lahkani, M.J.; Wang, S.; Urbański, M.; Egorova, M. Sustainable B2b E-Commerce and Blockchain-Based Supply Chain Finance. *Sustainability* **2020**, *12*, 3968. [CrossRef]
14. Chen, J.; Cai, T.; He, W.; Chen, L.; Zhao, G.; Zou, W.; Guo, L. A Blockchain-Driven Supply Chain Finance Application for Auto Retail Industry. *Entropy* **2020**, *22*, 95. [CrossRef]
15. Tsai, C.H.; Hung, C.W.; Su, P.C. Secure Authentication Scheme for an Agricultural Supply Chain Finance Environment. *Manag. Rev.* **2017**, *36*, 139–154. [CrossRef]
16. Du, M.; Chen, Q.; Xiao, J.; Yang, H.; Ma, X. Supply Chain Finance Innovation Using Blockchain. *IEEE Trans. Eng. Manag.* **2020**, *67*, 1045–1058. [CrossRef]
17. Dalal, D.; Yong, S.; Lewis, A. *The Future Is Here—Project Ubin: Sgd on Distributed Ledger*; Monetary Authority of Singapore and Deloitte: Singapore, 2017.
18. Guo, Y.; Liang, C. Blockchain Application and Outlook in the Banking Industry. *Financ. Innov.* **2016**, *2*, 24. [CrossRef]
19. Chang, S.E.; Chen, Y. When Blockchain Meets Supply Chain: A Systematic Literature Review on Current Development and Potential Applications. *IEEE Access* **2020**, *8*, 62478–62494. [CrossRef]

20. McWaters, R.J. *The Future of Financial Infrastructure: An Ambitious Look at How Blockchain Can Reshape Financial Services*; World Economic Forum: Colonie, Switzerland, 2016.

21. Lee, H. *Whitepaper 2.0 on Distributed Ledger Technology*; Hong Kong Monetary Authority: Hong Kong, China, 2017. Available online: https://www.hkma.gov.hk/media/eng/doc/key-functions/fininical-infrastructure/infrastructure/20171025e1a1.pdf (accessed on 11 March 2023).

22. Fnconn Financial. Available online: https://www.fnconn.com/index/Page/index/catid/2.html (accessed on 5 January 2022).

23. McWaters, R.J.; Bruno, G.; Lee, A.; Blake, M. *The Future of Financial Services: How Disruptive Innovations Are Reshaping the Way Financial Services Are Structured, Provisioned and Consumed*; World Economic Forum: Colonie, Switzerland, 2015.

24. P2p Lending. Available online: https://zh.wikipedia.org/wiki/2018%E5%B9%B4%E4%B8%AD%E5%9C%8B%E7%B6%B2%E7%B5%A1%E5%80%9F%E8%B2%B8%E5%B9%B3%E5%8F%B0%E9%9B%86%E9%AB%94%E5%80%92%E9%96%89%E4%BA%8B%E4%BB%B6 (accessed on 6 May 2021).

25. Yang, A.; Lee, V. The Development and Restriction of Virtual Currency in Taiwan from a Legal Perspective—Taking Bitcoin as an Example. In *Law and Development of Fintech*; Tseng, C.-Y., Ku, G., Eds.; Wu Nan: Taipei, Taiwan, 2017; pp. 63–73.

26. Tsang, C.-Y. Implications and Challenges of the Use of Blockchain for Financial Regulation. *Taiwan Law Rev.* **2017**, *267*, 136–152.

27. Huertas, J.; Liu, H.; Robinson, S. Eximchain: Supply Chain Finance Solutions on a Secured Public, Permissioned Blockchain Hybrid. In *Eximchain White Paper*; Eximchain: Central Region, Singapore, 2018; Available online: https://coinpaprika.com/storage/cdn/whitepapers/10568403.pdf (accessed on 1 March 2023).

28. Treiblmaier, H. The Token Economy as a Key Driver for Tourism: Entering the Next Phase of Blockchain Research. *Ann. Tour. Res.* **2021**, *91*, 103177. [CrossRef]

29. ERC-20 Token Standard. Available online: https://ethereum.org/en/developers/docs/standards/tokens/erc-20/ (accessed on 23 January 2023).

30. ERC-721 Non-Fungible Token Standard. Available online: https://ethereum.org/en/developers/docs/standards/tokens/erc-721/ (accessed on 23 January 2023).

31. Haber, S.; Stornetta, W.S. How to Time-Stamp a Digital Document I. *J. Cryptol.* **1991**, *3*, 99–111. [CrossRef]

32. Bayer, D.; Haber, S.; Stornetta, W.S. Improving the Efficiency and Reliability of Digital Time-Stamping. In *Sequences II: Methods in Communication, Security and Computer Science*; Springer: New York, NY, USA, 1993; pp. 329–334.

33. Ghosh, A.; Gupta, S.; Dua, A.; Kumar, N. Security of Cryptocurrencies in Blockchain Technology: State-of-Art, Challenges and Future Prospects. *J. Netw. Comput. Appl.* **2020**, *163*, 102635. [CrossRef]

34. Zheng, Z.; Xie, S.; Dai, H.-N.; Chen, X.; Wang, H. Blockchain Challenges and Opportunities: A Survey. *Int. J. Web Grid Serv.* **2018**, *14*, 352–375. [CrossRef]

35. Monrat, A.A.; Schelén, O.; Andersson, K. A Survey of Blockchain from the Perspectives of Applications, Challenges, and Opportunities. *IEEE Access* **2019**, *7*, 117134–117151. [CrossRef]

36. Khezr, S.; Moniruzzaman, M.; Yassine, A.; Benlamri, R. Blockchain Technology in Healthcare: A Comprehensive Review and Directions for Future Research. *Appl. Sci.* **2019**, *9*, 1736. [CrossRef]

37. Chod, J.; Trichakis, N.; Tsoukalas, G.; Aspegren, H.; Weber, M. On the Financing Benefits of Supply Chain Transparency and Blockchain Adoption. *Manag. Sci.* **2020**, *66*, 4378–4396. [CrossRef]

38. Attaran, M. Blockchain Technology in Healthcare: Challenges and Opportunities. *Int. J. Healthc. Manag.* **2020**, *15*, 70–83. [CrossRef]

39. Gorkhali, A.; Chowdhury, R. Blockchain and the Evolving Financial Market: A Literature Review. *J. Ind. Integr. Manag.* **2021**, *7*, 47–81. [CrossRef]

40. Chang, A.; El-Rayes, N.; Shi, J. Blockchain Technology for Supply Chain Management: A Comprehensive Review. *Fintech* **2022**, *1*, 191–205. [CrossRef]

41. Trollman, H.; Garcia-Garcia, G.; Jagtap, S.; Trollman, F. Blockchain for Ecologically Embedded Coffee Supply Chains. *Logistics* **2022**, *6*, 43. [CrossRef]

42. Shardeum Community. Testnet vs. Mainnet: What's the Difference? Available online: https://shardeum.org/blog/testnet-vs-mainnet/#What_is_a_Testnet (accessed on 19 February 2023).

43. Goerli Testnet. Available online: https://goerli.net/ (accessed on 19 February 2023).

44. Schoedon, A. The Görli Testnet Proposal—A Call for Participation. 2018. Available online: https://dev.to/5chdn/the-grli-testnet-proposal---a-call-for-participation-58pf (accessed on 19 February 2023).

45. Szabo, N. Smart Contracts: Building Blocks for Digital Markets. *EXTROPY J. Transhumanist Thought* **1996**, *18*, 28.

46. Wang, S.; Yuan, Y.; Wang, X.; Li, J.; Qin, R.; Wang, F.-Y. An Overview of Smart Contract: Architecture, Applications, and Future Trends. In Proceedings of the 2018 IEEE Intelligent Vehicles Symposium (IV), Changshu, China, 21 October 2018.

47. Zou, W.; Lo, D.; Kochhar, P.S.; Le, X.-B.D.; Xia, X.; Feng, Y.; Chen, Z.; Xu, B. Smart Contract Development: Challenges and Opportunities. *IEEE Trans. Softw. Eng.* **2019**, *47*, 2084–2106. [CrossRef]

48. Solidity. Available online: https://docs.soliditylang.org/en/v0.8.18/ (accessed on 19 February 2023).

49. Gupta, P.; Tham, T.M. Chapter 7. Blockchain and Distributed Ledger Technology 2.0. In *Fintech*; De Gruyter: Berlin, Germany; Boston, MA, USA, 2019; pp. 105–118.

50. Swan, M. *Blockchain: Blueprint for a New Economy*; O'Reilly Media, Inc.: Sevastopol, CA, USA, 2015.

51. Alharby, M.; Aldweesh, A.; van Moorsel, A. Blockchain-Based Smart Contracts: A Systematic Mapping Study of Academic Research (2018). In Proceedings of the 2018 International Conference on Cloud Computing, Big Data and Blockchain (ICCBB), Fuzhou, China, 15–17 November 2018.

52. Li, J.; Kassem, M. Applications of Distributed Ledger Technology (DLT) and Blockchain-Enabled Smart Contracts in Construction. *Autom. Constr.* **2021**, *132*, 103955. [CrossRef]

53. Dolgui, A.; Ivanov, D.; Potryasaev, S.; Sokolov, B.; Ivanova, M.; Werner, F. Blockchain-Oriented Dynamic Modelling of Smart Contract Design and Execution in the Supply Chain. *Int. J. Prod. Res.* **2019**, *58*, 2184–2199. [CrossRef]

54. Su, S.; Wang, K.; Kim, H.S. Smartsupply: Smart Contract Based Validation for Supply Chain Blockchain. In Proceedings of the 2018 IEEE International Conference on Internet of Things (iThings) and IEEE Green Computing and Communications (GreenCom) and IEEE Cyber, Physical and Social Computing (CPSCom) and IEEE Smart Data (SmartData), Halifax, NS, Canada, 30 July–3 August 2018.

55. Prause, G. Smart Contracts for Smart Supply Chains. *IFAC-PapersOnLine* **2019**, *52*, 2501–2506. [CrossRef]

56. Tolmach, P.; Li, Y.; Lin, S.-W.; Liu, Y.; Li, Z. A Survey of Smart Contract Formal Specification and Verification. *ACM Comput. Surv. (CSUR)* **2021**, *54*, 1–38. [CrossRef]

57. Liu, Z. Literature Review of Supply Chain Finance Based on Blockchain Perspective. *Open J. Bus. Manag.* **2021**, *9*, 419–429. [CrossRef]

58. Ghosh, B.; Paparas, D. Is There Any Pattern Regarding the Vulnerability of Smart Contracts in the Food Supply Chain to a Stressed Event? A Quantile Connectedness Investigation. *J. Risk Financial Manag.* **2023**, *16*, 58. [CrossRef]

59. Openzeppelin. Available online: https://www.openzeppelin.com/ (accessed on 19 February 2023).

60. Mougayar, W. Tokenomics—A Business Guide to Token Usage, Utility and Value. *Startup Manag.* **2017**. Available online: http://startupmanagement.org/2017/06/10/tokenomics-a-business-guide-to-token-usage-utility-and-value/ (accessed on 1 March 2023).

61. Wankmüller, C.; Pulsfort, J.; Kunovjanek, M.; Polt, R.; Craß, S.; Reiner, G. Blockchain-Based Tokenization and Its Impact on Plastic Bottle Supply Chains. *Int. J. Prod. Econ.* **2023**, *257*, 108776. [CrossRef]

62. Tian, Y.; Lu, Z.; Adriaens, P.; Minchin, R.E.; Caithness, A.; Woo, J. Finance Infrastructure through Blockchain-Based Tokenization. *Front. Eng. Manag.* **2020**, *7*, 485–499. [CrossRef]

63. Smith, J.; Vora, M.; Benedetti, H.; Yoshida, K.; Vogel, Z. Tokenized Securities and Commercial Real Estate. *Urban Econ.* **2019**. [CrossRef]

64. Baum, A. Tokenization—The Future of Real Estate Investment? *J. Portf. Manag.* **2021**, *47*, 41–61. [CrossRef]

65. Barbereau, T.; Sedlmeir, J.; Smethurst, R.; Fridgen, G.; Rieger, A. Tokenization and Regulatory Compliance for Art and Collectibles Markets: From Regulators' Demands for Transparency to Investors' Demands for Privacy. In *Blockchains and the Token Economy: Theory and Practice*; Springer: Cham, Germany, 2022; pp. 213–236.

66. Bamakan, S.M.H.; Nezhadsistani, N.; Bodaghi, O.; Qu, Q. Patents and Intellectual Property Assets as Non-Fungible Tokens; Key Technologies and Challenges. *Sci. Rep.* **2022**, *12*, 2178. [CrossRef]

67. Sazandrishvili, G. Asset Tokenization in Plain English. *J. Corp. Account. Financ.* **2020**, *31*, 68–73. [CrossRef]

68. Eip-1132: Extending Erc20 with Token Locking Capability. Available online: https://eips.ethereum.org/EIPS/eip-1132 (accessed on 23 January 2023).

69. The Test Website of SCF. Available online: http://django-env.eba-ktqcywad.us-west-2.elasticbeanstalk.com/ (accessed on 15 January 2023).

70. Savelyev, A. Contract Law 2.0: 'Smart'contracts as the Beginning of the End of Classic Contract Law. *Inf. Commun. Technol. Law* **2017**, *26*, 116–134. [CrossRef]

71. Wood, G. Ethereum: A Secure Decentralised Generalised Transaction Ledger. *Ethereum Proj. Yellow Paper.* **2014**, *151*, 1–32. Available online: https://files.gitter.im/ethereum/yellowpaper/VIyt/Paper.pdf (accessed on 1 March 2023).

72. Ethereum Average Gas Price Chart. Available online: https://etherscan.io/chart/gasprice (accessed on 1 December 2022).

73. Coinmarketcap. Available online: https://coinmarketcap.com/currencies/ethereum/historical-data/ (accessed on 1 December 2022).

74. Taiwan Smeg. Available online: https://www.smeg.org.tw/basic/?mode=detail&node=402 (accessed on 20 December 2022).

75. Templar, S. Supply Chain Finance Barometer 2018/2019. 2019. Available online: https://www.pwc.com/gx/en/deals/assets/scf-barometer-2018-2019-final-report.pdf (accessed on 19 February 2023).

76. Icc Global Survey on Trade Finance: Securing Future Growth. 2020. Available online: https://iccwbo.org/publication/global-survey/ (accessed on 19 February 2023).

77. Omran, Y.; Henke, M.; Heines, R.; Hofmann, E. *Blockchain-Driven Supply Chain Finance: Towards a Conceptual Framework from a Buyer Perspective*; IPSERA 2017: Budapest, Hungary, 10 April 2017. Available online: https://www.alexandria.unisg.ch/publications/251095 (accessed on 1 March 2023).