Intelligent contract design based on block chain in logistics management

Ping Yang*, Xiaolei Yang, Jianwei Li
State Grid Info-Telecom Great Power Science and Technology Co., LTD, Fujian, Fuzhou 350003, China
*a330499361@qq.com

Abstract. Although the logistics management has been improving, the information management is still stagnant. The renewal mechanism of logistics information is the basic requirement of logistics system, but it is more important to solve the trust problem of logistics information. It is good for improving the logistics management to design a verifiable information management mechanism which can help relevant participants establish trust relationships. This paper analyzes the development status of block chain technology, then a decentralized and self-verifiable system management program model is designed. Finally, the application is implemented based on the intelligent contract design of Ethereum block chain.

1. Introduction
In daily life, logistics management system is the basic information system. In the process of logistics management, some information systems can ensure the correct implementation of management [1-3]. As professor Martin Christopher said, the goal of logistics management is to coordinate all parties involved in providing products and services based on minimizing time and cost. Therefore, there must be a relevant scheme to coordinate these factors, which needs to consider the alternative method in some special cases. For example, there is a logistics transport \( T_m \) waiting for package \( P_i \) for days. If the logistics transport \( T_{m-1} \) fails to deliver the package to \( T_m \) before the deadline. The package will be paid by logistics \( T_{m-1} \), and it won’t be waived by the logistics \( T_m \) because the logistics information of this package will not be updated [4-6]. Therefore, the integrity and real-time update are the characteristics of the logistics management system information.

As is well-known, accurate information flow being the main foundation of logistics management. Centralized model is used to develop common methods of logistics management in the internet society where information can be accessed anywhere. At the same time, some innovative information management models are being developed, and some of them may be very practical in logistics management. Although centralized models are economical and practical, they are not always the best choice for a group of people who interact with each other and need mutual trust to achieve a common goal [7-8]. An example: user \( U_j \) can see the status information of his package \( P_i \), but this information is probably not enough for user \( U_j \) or the authenticity of the information is questionable.

Based on the trust problem of centralized model in logistics management system, this paper proposes a distributed model system of management information, which mainly requires that the information in the system can be verified and does not depend on the central mechanism or other external mechanisms.
And this technology has been applied to logistics management: tracking the entire supply chain of agricultural products based on RFID (radio frequency identification technology) and block chain technology, to obtain the credible information which storied from the produces to the sales period and processing, storage and all other intermediate stages [9]. As for the product supply chain, the primary sector and block chain technology solutions are very necessary. It will cause great social problems when the fake behaviour affects our health problems under the fake and inferior products are very popular in daily life. To solve this problem, Apte and Petrosky proposed a simple solution: track the charging process of a drug, understand and verify where and how the drug is traded. This is a good example to help us understand how block chain technology aggregates the factors which crucial to logistics [10]. The common characteristics these examples and our research is the ability to enhance the credibility of the current system by using a new tool that does not rely on any organizational mechanism. The aim of the study was to use Ethernet's main network to define a set of smart contracts that allow distributed applications. The program is executed on a public network of more than 20,000 active nodes with data privacy and confidentiality considerations. In this way, using the primary network has a greater advantage over using the private network.

The structure of this paper is as follows: In section 2, the working principle of block chain and Ethernet is explained; Section 3 is the preliminary preparation of the logistics management model; in section 4, the architecture of the scheme is introduced, including a case of delivery of goods. In section 5, data security issues related to cryptography are introduced, as well as how the scheme deals with system vulnerabilities and attacks. Finally, the conclusion is elaborated in section 6, and further summary and prospect are made.

2. Background

Block chain is like a ledger recording transactions, so it is considered as a distributed ledger technology in a narrow sense in many references. In block chain, when a transaction needs to be recorded, it will be appended to a new block. And block after block of data is linked chronologically into the total ledger (database), the latter block is generated based on the former block, as if all the data blocks are connected, called the "block chain". This ledger is maintained by multiple nodes on the network. Because of the difficulty of cracking cryptography, its data cannot be tampered with.

This means that nodes manage all transactions transparently, without having to delegate the transaction to a centralized authority such as a bank. With block chain, people who don't know each other can cooperate without an independent third party. In short, block chain is the source of trust. All the people in the block chain can participate in the network link, and each device can act as a node, and each node can obtain a complete copy of the database. Based on a set of consensus mechanism, nodes jointly maintain the whole block chain through competitive computing, and individual nodes fail or maliciously still work normally.

The main difference between block chain technology and the current economic management mode is that it does not rely on centralized management of Banks or head offices, eliminating the reliance on trust of third parties. Distributed ledger technology guarantees this trust because it is based on two properties: invariability of information and permanence of information. Technically, block chain is more complex than distributed ledger technology. The followings are the key points to understand the technique.

Data block: it is an ordered and grouped data set. It consists of two parts: the block header where the identifying information resides (the hash of the previous block and the hash of its own block always exist) and the transaction data set sorted by the Merkle tree structure. For the same block, the Merkle tree hash value must be the same.

Node: it is the computer which executes the block chain application. The program has an exact copy of the complete block chain and allows validation of transactions. Note that if someone wants to interact with the block chain, they don't need to act as active nodes (download all the blocks and act as validators). This pattern is called a client node.
Consensus mechanism: this process requires validation of all data blocks. Each node needs to be validated before the data block is added to the chain. Block chains require most nodes to validate each block, and if more than half of the nodes verify the block, it becomes part of the chain.

Miners and mining processes: when a node validates a data block, a high computational cost is required, which is reflected in the economic cost (power consumption or dedicated computer). However, block chain network needs to increase the calculation accuracy by increasing the number of active nodes. To encourage the addition of new nodes to the network, each node performing the work is paid accordingly. This incentive mechanism is the only way to create new nodes, and it is called mining, and the person who acts is the miner.

Figure 1 shows how the data block is linked to the previous one, considering the hash value of the block. Because the hash function is used on the block, the hash value of the block can be calculated. Each block can have multiple transactions and is part of a chain. As you can see from figure 1, the block header can be different from the block chain implementation section, but the implementation section has a reference to the previous block. For example, some version fields in bitcoins, hash values for trees, timestamps, or bitcoin. However, in the implementation of Ethereum, it also includes the limitation of commission or the field of commission use.

**Figure 1.** The block chain chain structure

Ethernet is another cryptocurrency besides bitcoin, which is called fuel by the developers. Ethernet coin is not real money. They are passwords used to process contracts and transactions. The main difference between a bitcoin block chain and an Ethereum block chain is the set of instructions available and the way they are managed. The former is created for a specific purpose, but the latter can already give it more flexibility so that developers can customize the goals. The solution in this paper is based on the implementation of Ethereum block chain, because it allows us to integrate intelligent contracts into logistics management through small-scale testing and distributed applications. The term "intelligent contract" was coined by Nick Szabp in 1994 in "computerised agreement to enforce the terms of the contract". At present, this concept has been applied to vending machines or offices accessed by electronic cards. As Buterin explains, it's all about implementing smart contracts to some extent. By incorporating block chain technology into the access control system, the system can increase the level of trust because all events in the chain are permanently registered and immutable objects. Because you can track specific events with trusted information, and because the entire information that can be registered is trusted, block chain provides integrity to the system. Therefore, in order to increase this feature and advantage in current transactions, more and more people are applying block chain in other fields.

To better understand Ethereum and the system in this article, we explain the proposal system in detail. In contrast to the implementation of bitcoin, Ethereum distinguishes between two types of users. All users are identified by a byte of address, and everyone can exchange the value and information.

The external owner account (EOA) is controlled by the private key ($sk_e$ represents the ether square private key) and there is no associated code.
The contract account (CA) is controlled by the code itself. The code can be modified to store, invoke other contract functions, or even create new contracts through transactions or messages.

Both EOA and CA have etheric balances that can perform transactions and send messages, as shown in figure 2. The difference between them depends on the signature. Messages are virtual objects that exist in the Ethereum virtual machine (EVM), which may contain the etheric balances or the data other than the sender and addressee addresses.

Both EOA and CA can send messages, and the receiver of the message can choose to receive the response. Messages are associated with the concept of functions. Messages cannot modify the internal storage of EOA or CA, and they can only read information.

Transactions in Ethereum may refer to the signature package containing messages and other data sent to the EOA. Instead of messages, transactions can update the internal storage of the CA or give Ethernet COINS to the EOA.

![Figure 2. The transactions between accounts](image)

In Ethereum, JSOM-RPC requests can interact with block chains. This means that Ethereum can give mobile and Web applications a simpler way to communicate, and the architecture proposed by the system requires this interaction. The following is a simplified trace of JSOM-RPC requests in the ether square.

```
bytes4 (keccak256 ("setName (string)")) = 0xc47f0027 (1)
EOAaddress= 0x43fc11,
CAaddress= 0xa8484f3,
Data [name] = “Paul” = 0x50 0x61 0x75 0x6c
Txdata= 0xc47f0027 · · · 5061756c · · · (2)
RPCrequest= {
“jsonrpc”: “2.0”,
“method”: “eth_sendTransaction”,
“params”: [{
“from”: “0x43fc11”,
“to”: “0xa8484f3”,
“data”: “0xc47f0027 · · · 5061756c · · · ”}],
“id” : 1}
```

The first step in instruction (1) is to identify the functions within the contract. Ethereum uses the first four bytes of the result calculated by the keccak256 hash function as the name of the contract function (input data type). The variable TXdata in instruction (1) adds the name of the function in order to associate the parameter (instruction 2) that needs to be received. At this point, the instruction code is associated with the address. The next instruction is to complete the JSON-RPC request and provide the transaction method (eth_sendTransaction method), the request parameters (source, target, and data), and the identity number (at this point, the value is 1).

### 3. Preliminary work

The modeling process of this scheme analyzes most of the work related to logistics management. In this section, the relevance between the baggage management system of the airport and the logistics management system will be explained. Figure 3 shows a simplified conceptual model of airport consignment management, which lays a foundation for the logistics management model studied in this
paper. The work is divided into four stages, of which the concept of three stages will be applied to the modeling of this paper (stages A, C and D in figure 3), as explained below:

A. Registration. When checking in their luggage, passengers are required to show identification for registration, to be associated with their luggage. The registered identity information is stored in the system, and then the luggage is labelled (one per bag) by the staff. After that, the system matches each luggage tag to the passenger.

B. Online verification. Passengers can check in their luggage at the airport baggage check point and verify their identity there.

C. Baggage Delivery Point. The verification number matches the NFC.

D. Baggage transfer point. At each control point, baggage status information is updated until the baggage is received at the baggage claim area at the destination airport.

Figure 3. Management process of baggage check at the airport
In the baggage consignment model, not only is a set of control points defined, but also a passenger and baggage registry are designed. Moreover, the modelling method of this problem can be considered as a logistics problem (see figure), and the research results of this paper can be applied in this field.

![Mapping relationship between baggage consignment management system and logistics management system](image)

**Figure 4.** Mapping relationship between baggage consignment management system and logistics management system

Figure 4 shows the correspondence between the baggage consignment system and the logistics management model. Each control point can act as an intermediary. The receiver and sender are usually the same person or EOA. Steps include: in the register, the person and baggage must be registered. Everyone has an address. This process is equivalent to converting them to EOAs. And the person's basic information must be registered in order to be associated with the luggage. Then, directly into the baggage delivery phase, EOA places baggage in a trusted intermediary at the first control point, which performs the operation through an intelligent contract. The status of the package is updated during different control points. Finally, the recipient of the luggage is the passenger himself. When passengers want to retrieve their luggage, they must verify that it meets expectations, thus ending the cycle in the smart contract.

4. Overall plan

This section proposes a concept design of intelligent contract based on block chain, which can give logistics system a greater degree of trust. Its main purpose is to provide more traceability throughout the baggage management process. Passengers can be managed through each of these intermediate control points when they deposit their luggage at the delivery point and collect their luggage at the destination airport. To achieve this goal, an intelligent contract is designed to register passengers and luggage and update the status of luggage in real time. For problems that require some private information to be stored in one of the smart contracts, this can be addressed by an external agent that sends encrypted data to the smart contract.

If there are many nodes in the network, the strength of consensus will increase. If there are 10,000 active nodes in the network, then 5001 nodes should validate each new block as part of the chain. However, in an internal network, such several nodes are unthinkable. For example, an airport environment can maintain several active nodes (if considered internally), such as four or five nodes, but not 1,000 nodes. In this case, the block is accepted as part of the chain only if three or four nodes agree on it, and it is also more likely to break than in 5,000 nodes. Of course, if the survey applies to national or international airport networks, private networks will take effect. In Spain, there are 48 active airports,
each with an internal network of four nodes, which will have 192 nodes. The internal network will now be a good choice because it will become more complex to include malicious blocks.

The packaging contract stores the package's main information and the address of its owner. When the package is delivered to the recipient, ownership changes and information is updated.

This information will be stored for a period for caching. Depending on the usage, the information may vary, but the owner and package address are the same. The smart contract allows two basic functions to be performed: one is to record and store the data of the package; the other is to record the relationship between the owner and the package. No one can update the status of a package unless authorized. For this function, you can specify permissions to perform this function (authorization) on a specific package that has been developed. The contract has been deployed on a network with the address as follows:

\[
0x574d573bdf25307088d8e9c27e47c467c0fe838e
\] (1)

User contracts contain common information (name, contact...) and personal information (financial address, identification...). As stated in the above contract, at least one legal identification number, such as the passport number of the airport, is required to connect an individual to an external owner's account. This intelligent contract has the basic functions of reading and writing in memory and storage. The contract has been deployed in the network and the address is as follows:

\[
0x0972fe92f8bfF05D1aabFAe0a5449e04836A40
\] (2)

Supply contracts allow control over the sender and receiver of packages and all intermediate accounts. Users and contract packages are simply stored as information with different policy access, but the smart contract handles the state of each package and its relationship to the list of owners, recipients, and mediators. It has three basic functions: generating new entries in the registry; Authorization, that is, permission to allow delegation to change the status of the package; Update the transporter, which allows you to update the state of the package (which is the contract executed by the middle tier and by calling the update state function). The contract has been deployed in the network and the address is as follows:

\[
0x2db2E6D741867f160A1E5a2359d18cB68268b49d
\] (3)

Smart contracts allow for complete control over the delivery of packages. They are not isolated contracts that simply store information, but interact (see figure 5). As mentioned above, each contract has several functions for keeping information up to date. Figure 5 is explained as follows:

u1. An external owner account (Auth) with the required permissions can register users on the system. EOA certification in baggage management systems is equivalent to everyone's ability to register new passengers.

p1. Auth can also record package information and its relationship with the package owner.

p2. Auth needs to define EOAs that can manage a particular package.

p3. Middle1 updates the state of the package.

p4. The state of the package is Middle2, which is the value delivered by the delivery contract.

p5. Alice confirmed that she received the package through the express contract. She is not authorized EOAs, but she is the recipient of the package and the system assumes authorized EOA.

d1. Bob registered and delivered his package. In that register, he needs to send the packet address and the recipient address (in this case, Alice's address). Middle1 updates the state of the package.

p3. When the update delivery function is triggered in the delivery contract, the status update function is automatically invoked in the package contract. This action allows you to update the information in the status package definition. The state of the package is the value of Middle1 for the delivery contract.

d3. Middle2 updates the state of the package.

p4. Alice confirmed that she received the package through the delivery contract. She is not authorized EOAs, but she is the recipient of the package and the system assumes authorized EOA.

p5. Information sent by Alice is stored in the contract package.
In this example, you can see that the interaction between contracts is quite simple, but the system needs to verify that all EOAs with the operation package information has enough permissions to perform the operation. The system is flexible and, in the current example, the sender and receiver are different people, but it can be used for baggage modelling without changing the contract code. If the recipient and the sender have the same address, the same person must collect the package at the end of the process.

In order to simplify the interpretation of the proposed system, the security of the system was not considered. All enquiries must be verified. In addition, each user's private information must be encrypted by an external proxy. There is also a smart contract that ACTS as a password library for validating information encrypted by an external proxy.

![Diagram of logistics entity interaction](Image)

**Figure 5.** Interaction instance of logistics entity (goods sent from Bob to Alice)

### 5. Safety analysis

This section focuses on security and provides an etheric space system for handling sensitive data. The limitations of how a language or system manages keys are addressed in the following sections.

Considering the limitations of unifying the language and creating exactly the same libraries inside and outside the etheric space, the maximum number data type managed uniformly is an unsigned integer with 256 bits. Due to the size limit, the number of elliptic ciphers used in several curves is less than this limit. For example, `scep256k1` curves are appropriate for $F_p$: $p = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$. However, there are also curves that work below the boundary, which can be parameterized with regular data types (or implemented at different times from basic operations). For example, the M-211 curve applies to $p = 2^{211} - 3$, or the curve M-2222 is used for $p = 2^{222} - 117$. Instead, there are other curves with finite fields of more than $2^{256}$ elements that cannot be interpreted by the language's basic data types. Asymmetric cryptography based on elliptic curves has been discarded, and the 256-bit key will become insecure and require an infinitely extensible array. It is possible to create a library that handles large numbers such as large integers in Java. For possibility, if more code is executed, GAS consumption also increases, resulting in an increase in the cost of its resolution based on the number and complexity of each operation performed.

Key management is divided into two parts: the public key used to encrypt the user's private information is developed through intelligent contract as a key store; all information is publicly accessible relative to the network, so encryption and decryption of the data has been delegated to external agents.
According to the example in figure 5, when Bob wants to send a package to Alice, the mediator needs to know where to extract the package and where to deliver it. Since the information stored in the user contract is private, it needs to be encrypted (no one should know anyone else's address unless necessary). To do this, the external agent needs to request private information from Bob and Alice to provide it to each broker that interacts in package management. In this case, both middle1 and middle2 can use the public key PKS stored in this contract to verify the authenticity of the information, and check whether Bob and Alice have signed the address information. The contract has been deployed in the network and its address is:

\[ \text{4513681977907369371490adcbdaedecbfdefefdex}) \]

(4)

In addition to the advantages and limitations of this technology, malicious behaviour is also considered in the design of the system. To ensure the confidentiality of the data stored in the network within each node of the network, it is necessary to encrypt it. Therefore, in order to ensure the privacy of information, elliptic algorithm is adopted as an encryption method. These contracts can be used regardless of whether EOA is part of the logistics system as they are publicly accessible. In this case, the system must provide authenticity. When EOA interacts with contracts and saves personal information, other users must be able to verify its authenticity. Therefore, information needs to be not only encrypted, but also signed. Fortunately, Ethernet provides a pre-compiled contract that can be accessed from anywhere as a native function. This function allows you to get the \( pk_E \) based on the EOA signature without implementing ECDSA. It not only reduces effort, but also reduces GAS consumption when validating information. Directive \( \text{erc4000}(\text{ECDSARECOVER}) \) has a fixed cost of 3,000 GAS units, but it performs less than A contract that USES the basic directive to do the same thing. There are three steps to generate an address in Ethernet (the symbol \( B_x \) represents a set of \( x \) bytes):

1. Select 32-byte private key in the range:

\[ [1, \text{secp256kl}_n^{-1}] \]  

(5)

2. Calculate a 64-bit public key using ECDSA, as shown below:

\[ \text{ECDSAPUBKEY}(sk_E \in B_{32}) = pk_E \in B_{64} \]  

(6)

3. Generates the latest 20-byte address with the following results:

\[ \text{keccak256}( pk_E [12 : 32] = EOA \text{address}) \]  

(7)

When EOA wants to sign a message, it needs to use its \( sk_E \) on the hash of the message to obtain a 65-byte 3-uple, which will be used as the parameter of the ECDSARECOVER function:

\[ \text{hash} \equiv \text{keccak 256(message) } \]  

(8)

\[ \text{signature} \equiv \text{ECDSASIGN } (\text{hash} \in B_{32}, sk_E \in B_{32}) = (v \in B_{32}, r \in B_{32}, s \in B_{32}) \]  

(9)

\[ pk_E \equiv \text{ECDSARECOVER \ ER(hash, v, r, s) } \]  

(10)

If you apply the last step of EOA address generation (3), the result is the address of the message signer. Therefore, there are two pairs of keys in the system, one set (\( pk_s, sk_s \)) for encryption and
decryption of the E1Gama1 library, and the other set (pkE, skE) for signing and validating EOAs using precompiled contracts. In conclusion, the confidentiality and authenticity of the system depend on the security and anti-attack ability of elliptic curve cryptography.

6. Summary
Based on the intelligent contract of Ethereum, this paper designs an information security management mechanism of logistics management. The scheme has strong scalability and adaptability, and can be easily deployed to different scale application scenarios. Preliminary application experiments show that the scheme has certain flexibility, can adapt to the rapid growth of the number of users, and enhances the reliability of logistics information management application. In addition, it increases the reduction of direct resources, such as application servers or storage space for backups, because blockchain is inherently redundant.

Acknowledgments
Author's brief introduction: Yang Ping (1978- ), male (Han nationality), Songxi County, Fujian Province, undergraduate, main research direction: computer software.

References
[1] Lefei Li; Shuming Tang. An Artificial Emergency-Logistics-Planning System for Severe Disasters, IEEE Intelligent Systems, 2008, 23 (4): 86 - 88.
[2] Shuang Li; Nengmin Wang; Tao Jia; Zhengwen He; Huigang Liang. Multiobjective Optimization for Multiperiod Reverse Logistics Network Design, IEEE Transactions on Engineering Management, 2016, 63 (2): 223 - 236.
[3] Mengke Yang; Movahedipour Mahmood; Xiaoguang Zhou; Salam Shafaq; Latif Zahid. Design and implementation of cloud platform for intelligent logistics in the trend of intellectualization, China Communications, 2017, 14 (10): 180 - 191.
[4] S. Keating; K. Huff. Managing risk in the new supply chain [performance-based logistics, Engineering Management Journal, 2005, 15 (1): 24 - 27.
[5] Uday Venkatadri; Kasinadhuni Shyama Krishna; M. Ali Ülkü. On Physical Internet Logistics: Modeling the Impact of Consolidation on Transportation and Inventory Costs, IEEE Transactions on Automation Science and Engineering, 2016, 13 (4): 1517 - 1527.
[6] Matthew J. Liberatore; Bruce Pollack-Johnson. Improving Project Management Decision Making by Modeling Quality, Time, and Cost Continuously, IEEE Transactions on Engineering Management, 2013, 60 (3): 518 - 528.
[7] Hang Xu; Rang Xu; Qingtai Ye. Optimization of unbalanced multi-stage logistics systems based on prifer number and effective capacity coding, Tsinghua Science and Technology, 2006, 11 (1): 96 - 101.
[8] Xiangtong Qi. Coordinated Logistics Scheduling for In-House Production and Outsourcing, IEEE Transactions on Automation Science and Engineering, 2008, 5 (1): 188 - 192.
[9] Aqsa Naeem; Ali Shabbir; Naveed Ul Hassan; Chau Yuen; Ayaz Ahmad; Wayes Tushar. Understanding Customer Behavior in Multi-Tier Demand Response Management Program, IEEE Access, 2015, 3: 2613 - 2625.
[10] Leslie P. Martinich. Excellent execution: production and operations management, IEEE Engineering Management Review, 2014, 42 (4): 13 - 15.