

SYSTEMS AND METHODS FOR DISTRIBUTED-LEDGER BASED INTERCOMPANY NETTING

RELATED APPLICATIONS

[0001] This application claims priority to, and the benefit of, U.S. Provisional Patent Application Ser. No. 62/686,131, filed June 18, 2018, the disclosure of which is hereby incorporated, by reference, in its entirety.

BACKGROUND OF THE INVENTION

1. Field of the Invention

[0002] The present disclosure generally relates to systems and methods for distributed-ledger based intercompany netting.

2. Description of the Related Art

[0003] Intercompany Netting is a process that helps in optimizing capital, mitigating risk, and lowering transaction cost. The process, however, remains time consuming, involves multiple steps, and is dependent on multiple parties.

SUMMARY OF THE INVENTION

[0004] Systems and methods for distributed-ledger based intercompany netting are disclosed. In one embodiment, in an information processing apparatus comprising at least one computer processor, a method for distributed-ledger based intercompany netting may include: (1) receiving, from a first entity within an organization, a deposit of a first amount of funds to a first account; (2) tokenizing the first amount of funds and writing the first amount of

funds to a first token wallet for the first entity on a distributed ledger; (3) receiving, from a second entity within an organization, a deposit of a second amount of funds to a second account; (4) tokenizing the second amount of funds and writing the second amount of funds to a second token wallet for the second entity on the distributed ledger; and (5) executing a transaction involving a transfer of a transaction amount from the first entity to the second entity by deducting the transaction amount from a token balance in the first wallet and adding the transaction amount to a token balance in the second wallet.

[0005] In one embodiment, the first amount of funds and the second amount of funds may be tokenized with a one-to-one correspondence with a base currency.

[0006] In one embodiment, the method may further include applying a restriction to the first amount of funds in the first account after the first amount of funds is tokenized, wherein the restriction restricts access to the first amount of funds in the account.

[0007] In one embodiment, the method may further include receiving a surrender of a surrender token amount from the first entity; deducting the surrender token amount from the first wallet; and releasing the restriction on the surrender amount of funds in the first account.

[0008] In one embodiment, the tokens in the first wallet may correspond to the surrender amount are destroyed or deleted in response to the deduction.

[0009] In one embodiment, the deduction for the first wallet may be written to the distributed ledger.

[0010] In one embodiment, the method may further include reconciling the first account and the second account based on the balances in the first wallet and the second wallet, respectively.

[0011] In one embodiment, the step of deducting the transaction amount from a token balance in the first wallet and adding the transaction amount to a token balance in the second wallet may be performed by a smart contract.

[0012] In one embodiment, the first entity, the second entity, and the financial institution may be nodes in a distributed ledger network and each maintain a copy of the distributed ledger.

[0013] In one embodiment, the first account and/or the second account may be an escrow account, a treasury omnibus account, etc.

[0014] According to another embodiment, a system for distributed-ledger based intercompany netting may include a first entity within an organization; a second entity within an organization; and a financial institution that maintains a first account for the first entity, and a second account for the second entity. The financial institution may receive a deposit of a first amount of funds to the first account from the first entity; may tokenize the first amount of funds and writes the first amount of funds to a first token wallet for the first entity on a distributed ledger; may receive a deposit of a second amount of funds to the second account from the second entity; may tokenize the second amount of funds and writes the second amount of funds to a second token wallet for the second entity on the distributed ledger; and may write a transaction involving a transfer of a transaction amount from the first entity to the second entity to the distributed ledger. A smart contract executed by the distributed ledger may

deduct the transaction amount from a token balance in the first wallet and adds the transaction amount to a token balance in the second wallet.

[0015] In one embodiment, the first amount of funds and the second amount of funds may be tokenized with a one-to-one correspondence with a base currency.

[0016] In one embodiment, the financial institution may apply a restriction to the first amount of funds in the first account after the first amount of funds is tokenized, wherein the restriction restricts access to the first amount of funds in the account.

[0017] In one embodiment, the financial institution may receive a surrender of a surrender token amount from the first entity, may deduct the surrender token amount from the first wallet, and may release the restriction on the surrender amount of funds in the first account.

[0018] In one embodiment, the tokens in the first wallet corresponding to the surrender amount may be destroyed or deleted in response to the deduction.

[0019] In one embodiment, a smart contract may write the deduction for the first wallet to the distributed ledger.

[0020] In one embodiment, the financial institution may reconcile the first account and the second account based on the balances in the first wallet and the second wallet, respectively.

[0021] In one embodiment, the first entity, the second entity, and the financial institution may be nodes in a distributed ledger network and each maintains a copy of the distributed ledger.

[0022] In one embodiment, the first account and/or the second account may be an escrow account.

BRIEF DESCRIPTION OF THE DRAWINGS

[0023] For a more complete understanding of the present invention, the objects and advantages thereof, reference is now made to the following descriptions taken in connection with the accompanying drawings in which:

[0024] Figure 1 depicts a system for distributed-ledger based intercompany netting according to one embodiment; and

[0025] Figure 2 depicts a method for distributed-ledger based intercompany netting according to one embodiment.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0026] Embodiments are directed to the use of distributed ledgers, such as Blockchain distributed ledgers, for intercompany netting.

[0027] Distributed ledgers provide a peer-to-peer network with no central authority. They are secure and use cryptography to determine valid parties and transactions. Each step in a transaction is immutable as it is digitally signed with a strong audit trail, and each piece of data in a transaction can be given a unique fingerprint, such as a hash.

[0028] In embodiments, distributed ledgers, such as Blockchain distributed ledgers, may be used for intercompany netting. For example, fiat cash may be converted to corporate coins, or tokens. The corporate coins may be minted by a bank, and may be credited to a subsidiary wallet at the back of fiat currency collateral from a treasury. The subsidiaries may then settle their intercompany transactions using their corporate coins on a peer-to-peer basis. The subsidiaries may then redeem their corporate coins into any fiat currency through the bank.

[0029] Embodiments may improve transparency, eliminate steps, and may be automated and continuous.

[0030] Referring to Figure 1, a system for distributed-ledger based intercompany netting is disclosed according to one embodiment. System 100 may include organization 110 which may be any type of organization (e.g., public, private, etc.). Organization may have one or more subsidiaries, lines of business, entities, etc. 120₁, 120₂, ... 120_n. Organization 110 and subsidiaries 120₁, 120₂, ... 120_n may each have one or more account with financial institution 130. For example, financial institution 130 may maintain an account for organization 110 and subsidiaries 120₁, 120₂, ... 120_n and may issue tokens in exchange for deposits to the account.

[0031] In one embodiment, the account may be a treasury omnibus account that may hold cash from all the different entities against which tokens are issued. In another embodiment, the account may be an escrow account.

[0032] Treasury 140 may be provided, and may represent the corporate Treasury / HQ for organization 110.

[0033] Distributed ledger network 115, 125, 135, 145, etc. may be any suitable ledger, including, for example, Blockchain-based ledgers, Ethereum ledgers, etc. Organization 110, subsidiaries 120₁, 120₂, ... 120_n and financial institution 130 may each have a client (not shown) that may interact with distributed ledger network 115, 125, 135, 145 (e.g., write transactions to their respective copies of the distributed ledger).

[0034] In one embodiment, distributed ledger network 115, 125, 135, 145 may provide a smart contract platform, confidentiality (e.g., details of transactions may be private to the parties), security (e.g., advanced encryption, permissioned-only), and decentralization (e.g., no dependency on a central service). It may also provide digitalization of payments with real-time gross settlement capabilities, decentralized processing (e.g., distributed and resilient infrastructure with no single point of failure), settlement finality (e.g., final and irrevocable settlement of payment instructions with deterministic finality), and may optimize liquidity by implementing netting and gridlock resolution algorithms to maximize liquidity efficiency.

[0035] In addition, distributed ledger network 115, 125, 135, 145 may be provided with a zero-knowledge security layer, which may enable mass conservation and prevention of double-spend for shielded tokens without compromising decentralized nature of the ledger, and may use zk-SNARKs (zero-knowledge cryptographic proofs).

[0036] In one embodiment, smart contracts may execute based on the contents of distributed ledger network 115, 125, 135, 145. For example, smart contracts may function to increase or decrease organization 110 or subsidiary

120's tokens in response to a transaction that is written to distributed ledger network 115, 125, 135, 145, may cause financial institution 130 to issue cash to organization 110 or subsidiary 120 in response to a surrender of tokens, etc.

[0037] Embodiments may further include third party customers (not shown), third party suppliers (not shown), third party distributors (not shown), etc. These third parties may also deposit funds to an account, and may also access distributed ledger network 115, 125, 135, 145 to write transactions.

[0038] Referring to Figure 2, a method for distributed-ledger based intercompany netting is disclosed according to one embodiment.

[0039] In step 205, each entity in an organization (e.g., a subsidiary) may deposit funds in an account with a financial institution. In one embodiment, each entity may be a separate legal entity, lines of business, or any part of a larger organization that may have its own budget.

[0040] In one embodiment, a single account may be used for multiple entities in the organization; in another embodiment, each entity may have its own account.

[0041] In one embodiment, the account may be a treasury omnibus account that may hold cash from all the different entities against which tokens are issued. In another embodiment, the account may be an escrow account.

[0042] In step 210, the financial institution may issue token(s) for the deposited funds. In one embodiment, the tokens may be issued to have a one-to-one correspondence with the unit of currency deposited (e.g., one token for each dollar, etc.). Thus, if an entity deposits \$10,000, it may be issued 10,000

tokens. Other ratios and correspondences may be used as is necessary and/or desired.

[0043] In step 215, the financial institution may write the token deposits for each entity to a distributed ledger in a distributed ledger network. In one embodiment, the token deposit may be added to an existing token balance for each entity.

[0044] In one embodiment, the distributed ledger may be a Blockchain-based distributed ledger. Other suitable ledgers may be used as is necessary and/or desired.

[0045] In one embodiment, each entity may maintain a wallet on the distributed ledger, and the wallet may reflect the respective entity's token balance.

[0046] In one embodiment, after the funds are tokenized, a restriction may be associated with the corresponding funds to that they are identified as not available or otherwise restricted from being accessed, withdrawn, or transferred. Any suitable manner of restricting access to the tokenized funds may be used as is necessary and/or desired.

[0047] In step 220, a first organizational entity may write a transaction involving a second organizational entity to the distributed ledger. In one embodiment, the first entity may identify the source of the funds, the recipient of the funds, the amount, etc. In one embodiment, the transaction may identify the number of tokens that correspond to the amount of the transaction.

[0048] Any other information may be provided as is necessary and/or desired.

[0049] In step 225, a smart contract may execute the transaction and may take the appropriate action with the tokens. For example, if the transaction involves the first entity transferring funds to the second entity, the smart contract may debit the number of tokens in from the source of the funds (e.g., the first entity's wallet), and deposit tokens to the recipient of the funds (e.g., the second entity's wallet).

[0050] In one embodiment, the transaction may only involve the first entity, such as when the first entity surrenders tokens for cash. In one embodiment, when the tokens are surrendered, the tokens may be destroyed, deleted, etc., and any restrictions on the corresponding funds may be released. Thus, the entity may be able to access, withdraw, transfer, etc. funds from the account as desired.

[0051] In another embodiment, the smart contract may provide the corresponding funds to the surrendering entity's account in exchange for the surrender of the tokens.

[0052] In step 230, the accounts and the wallets may be reconciled to reflect one or more transaction. In one embodiment, the accounts may be reconciled periodically (e.g., daily, weekly, etc.), on demand, or as otherwise necessary and/or desired.

[0053] In one embodiment, smart contracts may compute interest based on deposit rates for the outstanding tokens, cause tokens to self-destruct, auto-

redeem to fiat cash based on an outstanding time period, and perform tax computations on token movements (e.g., withholding tax, etc.).

[0054] Embodiments may provide some or all of the following advantages and improvements: they may increase operating liquidity (e.g., operating liquidity trapped with subsidiaries centralized by design without sacrificing decentralized control over cash); increase efficiency (e.g., physical cash flows replaced with real time corporate coin/token transfers thereby eliminating process and infrastructure in-efficiencies); and simplify foreign exchange handling.

[0055] Although several embodiments have been disclosed, it should be recognized that these embodiments are not mutually exclusive, and features from one may be used with others.

[0056] The following documents are hereby incorporated, by reference, in their entireties: U.S. Patent Application Serial Nos. 15/903,215; 15/903,159; 15/475,846; 15/797,602; 62/414,398; 62/534,772 and PCT Application No. PCT/US17/59015.

[0057] Hereinafter, general aspects of implementation of the systems and methods of the invention will be described.

[0058] The system of the invention or portions of the system of the invention may be in the form of a “processing machine,” such as a general purpose computer, for example. As used herein, the term “processing machine” is to be understood to include at least one processor that uses at least one memory. The at least one memory stores a set of instructions. The instructions

may be either permanently or temporarily stored in the memory or memories of the processing machine. The processor executes the instructions that are stored in the memory or memories in order to process data. The set of instructions may include various instructions that perform a particular task or tasks, such as those tasks described above. Such a set of instructions for performing a particular task may be characterized as a program, software program, or simply software.

[0059] In one embodiment, the processing machine may be a specialized processor.

[0060] As noted above, the processing machine executes the instructions that are stored in the memory or memories to process data. This processing of data may be in response to commands by a user or users of the processing machine, in response to previous processing, in response to a request by another processing machine and/or any other input, for example.

[0061] As noted above, the processing machine used to implement the invention may be a general purpose computer. However, the processing machine described above may also utilize any of a wide variety of other technologies including a special purpose computer, a computer system including, for example, a microcomputer, mini-computer or mainframe, a programmed microprocessor, a micro-controller, a peripheral integrated circuit element, a CSIC (Customer Specific Integrated Circuit) or ASIC (Application Specific Integrated Circuit) or other integrated circuit, a logic circuit, a digital signal processor, a programmable logic device such as a FPGA, PLD, PLA or

PAL, or any other device or arrangement of devices that is capable of implementing the steps of the processes of the invention.

[0062] The processing machine used to implement the invention may utilize a suitable operating system. Thus, embodiments of the invention may include a processing machine running the iOS operating system, the OS X operating system, the Android operating system, the Microsoft Windows™ operating systems, the Unix operating system, the Linux operating system, the Xenix operating system, the IBM AIX™ operating system, the Hewlett-Packard UX™ operating system, the Novell Netware™ operating system, the Sun Microsystems Solaris™ operating system, the OS/2™ operating system, the BeOS™ operating system, the Macintosh operating system, the Apache operating system, an OpenStep™ operating system or another operating system or platform.

[0063] It is appreciated that in order to practice the method of the invention as described above, it is not necessary that the processors and/or the memories of the processing machine be physically located in the same geographical place. That is, each of the processors and the memories used by the processing machine may be located in geographically distinct locations and connected so as to communicate in any suitable manner. Additionally, it is appreciated that each of the processor and/or the memory may be composed of different physical pieces of equipment. Accordingly, it is not necessary that the processor be one single piece of equipment in one location and that the memory be another single piece of equipment in another location. That is, it is contemplated that the processor may be two pieces of equipment in two different physical locations. The two distinct pieces of equipment may be

connected in any suitable manner. Additionally, the memory may include two or more portions of memory in two or more physical locations.

[0064] To explain further, processing, as described above, is performed by various components and various memories. However, it is appreciated that the processing performed by two distinct components as described above may, in accordance with a further embodiment of the invention, be performed by a single component. Further, the processing performed by one distinct component as described above may be performed by two distinct components. In a similar manner, the memory storage performed by two distinct memory portions as described above may, in accordance with a further embodiment of the invention, be performed by a single memory portion. Further, the memory storage performed by one distinct memory portion as described above may be performed by two memory portions.

[0065] Further, various technologies may be used to provide communication between the various processors and/or memories, as well as to allow the processors and/or the memories of the invention to communicate with any other entity; i.e., so as to obtain further instructions or to access and use remote memory stores, for example. Such technologies used to provide such communication might include a network, the Internet, Intranet, Extranet, LAN, an Ethernet, wireless communication via cell tower or satellite, or any client server system that provides communication, for example. Such communications technologies may use any suitable protocol such as TCP/IP, UDP, or OSI, for example.

[0066] As described above, a set of instructions may be used in the processing of the invention. The set of instructions may be in the form of a program or software. The software may be in the form of system software or application software, for example. The software might also be in the form of a collection of separate programs, a program module within a larger program, or a portion of a program module, for example. The software used might also include modular programming in the form of object oriented programming. The software tells the processing machine what to do with the data being processed.

[0067] Further, it is appreciated that the instructions or set of instructions used in the implementation and operation of the invention may be in a suitable form such that the processing machine may read the instructions. For example, the instructions that form a program may be in the form of a suitable programming language, which is converted to machine language or object code to allow the processor or processors to read the instructions. That is, written lines of programming code or source code, in a particular programming language, are converted to machine language using a compiler, assembler or interpreter. The machine language is binary coded machine instructions that are specific to a particular type of processing machine, i.e., to a particular type of computer, for example. The computer understands the machine language.

[0068] Any suitable programming language may be used in accordance with the various embodiments of the invention. Illustratively, the programming language used may include assembly language, Ada, APL, Basic, C, C++, COBOL, dBase, Forth, Fortran, Java, Modula-2, Pascal, Prolog, REXX, Visual Basic, and/or JavaScript, for example. Further, it is not necessary that a single

type of instruction or single programming language be utilized in conjunction with the operation of the system and method of the invention. Rather, any number of different programming languages may be utilized as is necessary and/or desirable.

[0069] Also, the instructions and/or data used in the practice of the invention may utilize any compression or encryption technique or algorithm, as may be desired. An encryption module might be used to encrypt data. Further, files or other data may be decrypted using a suitable decryption module, for example.

[0070] As described above, the invention may illustratively be embodied in the form of a processing machine, including a computer or computer system, for example, that includes at least one memory. It is to be appreciated that the set of instructions, i.e., the software for example, that enables the computer operating system to perform the operations described above may be contained on any of a wide variety of media or medium, as desired. Further, the data that is processed by the set of instructions might also be contained on any of a wide variety of media or medium. That is, the particular medium, i.e., the memory in the processing machine, utilized to hold the set of instructions and/or the data used in the invention may take on any of a variety of physical forms or transmissions, for example. Illustratively, the medium may be in the form of paper, paper transparencies, a compact disk, a DVD, an integrated circuit, a hard disk, a floppy disk, an optical disk, a magnetic tape, a RAM, a ROM, a PROM, an EPROM, a wire, a cable, a fiber, a communications channel, a satellite transmission, a memory card, a SIM card, or other remote transmission,

as well as any other medium or source of data that may be read by the processors of the invention.

[0071] Further, the memory or memories used in the processing machine that implements the invention may be in any of a wide variety of forms to allow the memory to hold instructions, data, or other information, as is desired. Thus, the memory might be in the form of a database to hold data. The database might use any desired arrangement of files such as a flat file arrangement or a relational database arrangement, for example.

[0072] In the system and method of the invention, a variety of “user interfaces” may be utilized to allow a user to interface with the processing machine or machines that are used to implement the invention. As used herein, a user interface includes any hardware, software, or combination of hardware and software used by the processing machine that allows a user to interact with the processing machine. A user interface may be in the form of a dialogue screen for example. A user interface may also include any of a mouse, touch screen, keyboard, keypad, voice reader, voice recognizer, dialogue screen, menu box, list, checkbox, toggle switch, a pushbutton or any other device that allows a user to receive information regarding the operation of the processing machine as it processes a set of instructions and/or provides the processing machine with information. Accordingly, the user interface is any device that provides communication between a user and a processing machine. The information provided by the user to the processing machine through the user interface may be in the form of a command, a selection of data, or some other input, for example.

[0073] As discussed above, a user interface is utilized by the processing machine that performs a set of instructions such that the processing machine processes data for a user. The user interface is typically used by the processing machine for interacting with a user either to convey information or receive information from the user. However, it should be appreciated that in accordance with some embodiments of the system and method of the invention, it is not necessary that a human user actually interact with a user interface used by the processing machine of the invention. Rather, it is also contemplated that the user interface of the invention might interact, i.e., convey and receive information, with another processing machine, rather than a human user. Accordingly, the other processing machine might be characterized as a user. Further, it is contemplated that a user interface utilized in the system and method of the invention may interact partially with another processing machine or processing machines, while also interacting partially with a human user.

[0074] It will be readily understood by those persons skilled in the art that the present invention is susceptible to broad utility and application. Many embodiments and adaptations of the present invention other than those herein described, as well as many variations, modifications and equivalent arrangements, will be apparent from or reasonably suggested by the present invention and foregoing description thereof, without departing from the substance or scope of the invention.

[0075] Accordingly, while the present invention has been described here in detail in relation to its exemplary embodiments, it is to be understood that this disclosure is only illustrative and exemplary of the present invention and is made to provide an enabling disclosure of the invention. Accordingly, the

foregoing disclosure is not intended to be construed or to limit the present invention or otherwise to exclude any other such embodiments, adaptations, variations, modifications or equivalent arrangements.

CLAIMS

1. A method for distributed-ledger based intercompany netting involving a first entity within an organization and a second entity within the organization, and a financial institution comprising:

receiving, at the financial institution and from the first entity, a first deposit of a first amount of fiat currency funds to a first account for storing a fiat currency;

writing, by the financial institution and in response to the first deposit, a first token-funding transaction to a first copy of a distributed ledger maintained by the financial institution, wherein the first transaction indicates the first amount of fiat currency funds;

executing, in response to the writing of the first transaction, a tokenization smart contract tokenizes the first amount of fiat currency funds to a first amount of digital token funds and writes the first amount of digital token funds to a first token wallet for the first entity on the first copy of the distributed ledger maintained by the financial institution;

executing, in response to the tokenization of the first amount of fiat currency, a restriction smart contract, wherein the restriction smart contract applies a first restriction to the first amount of fiat currency funds in the first account, and wherein the first restriction restricts access to the first amount of fiat currency funds in the first account;

receiving, at the financial institution and from the second entity, a second deposit of a second amount of fiat currency funds to a second account for storing a fiat currency;

writing, by the financial institution and in response to the second deposit, a second token-funding transaction to the first copy of a distributed ledger

maintained by the financial institution, wherein the second transaction indicates the second amount of fiat currency funds;

executing, in response to the second transaction, the tokenization smart contract, wherein the tokenization smart contract tokenizes the second amount of fiat currency funds to a second amount of digital token funds and writes the second amount of digital token funds to a second token wallet for the second entity on the first copy of the distributed ledger maintained by the financial institution;

executing, in response to the tokenization of the second amount of fiat currency, the restriction smart contract, wherein the restriction smart contract applies a second restriction to the second amount of fiat currency funds in the second account, and wherein the second restriction restricts access to the second amount of fiat currency funds in the second account;

writing, by the first entity, an inter-entity transaction to a second copy of the distributed ledger maintained by the first entity, wherein the inter-entity transaction specifies a transfer amount of digital token funds, and wherein the transfer amount of digital token funds indicates an amount of digital token funds to be transferred from the first entity to the second entity;

executing, in response to the inter-entity transaction, a netting smart contract, wherein the netting smart contract determines the transfer amount of digital token funds, deducts the transfer amount from a token balance of the first token wallet, and adds the transfer amount to a token balance of the second token wallet; and

reconciling the first account and the second account based on the balances in the first token wallet and the second token wallet, respectively.

2. The method of claim 1, wherein the first amount of fiat currency funds and the second amount of fiat currency funds are tokenized to digital token funds with a one-to-one correspondence with the fiat currency funds.

3. The method of claim 1, further comprising:
receiving a surrender token amount from the first entity, wherein the surrender token amount comprises an amount of tokens from the first amount of digital token funds to be redeemed for a corresponding amount of fiat currency funds;
deducting, by the restriction smart contract, the surrender token amount from the first token wallet; and
partially releasing, by the restriction smart contract, the first restriction to the first amount of fiat currency funds in the first account, wherein the partially releasing releases a partial amount of the first amount of fiat currency equal to the corresponding amount of fiat currency funds.

4. The method of claim 3, wherein the restriction smart contract destroys or deletes the surrender token amount of tokens from the first token wallet.

5. The method of claim 3, wherein the deduction for the first token wallet is written to the distributed ledger.

6. The method of claim 1, wherein the first entity, the second entity, and the financial institution are nodes in a distributed ledger network and each maintain a copy of the distributed ledger.

7. The method of claim 1, wherein the first account or the second account comprises an escrow account.

8. The method of claim 1, wherein the first account or the second account comprises a treasury omnibus account.

9. A system for distributed-ledger based intercompany netting, comprising:

a first entity within an organization;

a second entity within the organization; and

a financial institution that maintains a first account for the first entity, and a second account for the second entity, wherein the first account and the second account are configured to store a fiat currency;

wherein:

the financial institution receives a deposit of a first amount of fiat currency funds to the first account from the first entity;

the financial institution writes, in response to the first deposit, a first token-funding transaction to a first copy of the distributed ledger maintained by the financial institution, wherein the first transaction indicates the first amount of fiat currency funds;

in response to the first transaction, a tokenization smart contract executes, wherein the tokenization smart contract tokenizes the first amount of fiat currency funds to a first amount of digital token funds and writes the first amount of digital token funds to a first token wallet for the first entity on the first copy of the distributed ledger maintained by the financial institution;

in response to the tokenization of the first amount of fiat currency, a restriction smart contract executes, wherein the restriction smart contract applies a first restriction to the first amount of fiat currency funds in the first account, and wherein the first restriction restricts access to the first amount of fiat currency funds in the first account;

the financial institution receives a deposit of a second amount of fiat currency funds to the second account from the second entity;

the financial institution writes, in response to the second deposit, a second token-funding transaction to the first copy of the distributed ledger maintained by the financial institution, wherein the second transaction indicates the second amount of fiat currency funds;

in response to the second transaction, the tokenizing smart contract tokenizes the second amount of fiat currency funds to a second amount of digital token funds and writes the second amount of digital token funds to a second token wallet for the second entity on the first copy of the distributed ledger maintained by the financial institution;

in response to the tokenization of the second amount of fiat currency, the restriction smart contract executes, wherein the restriction smart contract applies a second restriction to the second amount of fiat currency funds in the second account, and wherein the second restriction restricts access to the second amount of fiat currency funds in the second account;

the first entity writes an inter-entity transaction to a second copy of the distributed ledger maintained by the first entity, wherein the inter-entity transaction specifies a transfer amount of digital token funds, and wherein the transfer amount of digital token funds indicates an amount of digital token funds to be transferred from the first entity to the second entity; and

in response to the inter-entity transaction, a netting smart contract executes, wherein the netting smart contract determines the transfer amount of digital token funds, deducts the transfer amount from a token balance of the first token wallet, and adds the transfer amount to a token balance of the second token wallet; and

the financial institution reconciles the first account and the second account based on the balances in the first token wallet and the second token wallet, respectively.

10. The system of claim 9, wherein the first amount of fiat currency funds and the second amount of fiat currency funds are tokenized with a one-to-one correspondence with the fiat currency funds.

11. The system of claim 9, wherein:
the financial institution receives a surrender token amount from the first entity, wherein the surrender token amount comprises an amount of tokens from the first amount of digital token funds to be redeemed for a corresponding amount of fiat currency funds;

the restriction smart contract deducts the surrender token amount from the first token wallet; and

the restriction smart contract makes a partial release of the first restriction to the first amount of fiat currency funds in the first account, the partial release comprising releasing a partial amount of the first amount of fiat currency equal to the corresponding amount of fiat currency funds.

12. The system of claim 11, wherein the restriction smart contract destroys or deletes the surrender token amount of tokens from the first token wallet.

13. The system of claim 9, wherein the first entity, the second entity, and the financial institution are nodes in a distributed ledger network and each maintains a copy of the distributed ledger.

14. The system of claim 9, wherein the first account or the second account comprises an escrow account.