US 20200042982A1

(19) United States

(12) Patent Application Publication (10) Pub. No.: US 2020/0042982 A1

Snow (43) Pub. Date: Feb. 6, 2020

(54) **DIGITAL CONTRACTS IN BLOCKCHAIN ENVIRONMENTS**

(71) Applicant: **Factom**, Austin, TX (US)

(72) Inventor: **Paul Snow**, Austin, TX (US)

(73) Assignee: **Factom**, Austin, TX (US)

(21) Appl. No.: **16/116,966**

(22) Filed: **Aug. 30, 2018**

Related U.S. Application Data

(60) Provisional application No. 62/714,909, filed on Aug. 6, 2018.

Publication Classification

(51) **Int. Cl.**

G06Q 20/36	(2006.01)
H04L 9/06	(2006.01)
G06Q 20/06	(2006.01)
G06F 21/53	(2006.01)

(52) **U.S. Cl.**

CPC ***G06Q 20/367*** (2013.01); ***G06F 21/53*** (2013.01); ***G06Q 20/0658*** (2013.01); ***H04L 9/0637*** (2013.01)

(57) **ABSTRACT**

Digital or "smart" contracts execute in a blockchain environment. Any entity (whether public or private) may specify a digital contract via a contract identifier in a blockchain. Because there may be many digital contracts offered as services, the contract identifier uniquely identifies a particular digital contract offered by a vendor or supplier. The contract identifier may also uniquely identify a virtual machine that executes the digital contract. Virtual machines may thus be preassigned to execute particular digital contracts. Moreover, data records in a blockchain data layer may be generated that document execution of the digital contract by the virtual machine. The data records in the blockchain data layer may also document any transaction records of a cryptocoinage that is paid, redeemed, traded, or transferred according to the terms of the digital contract.

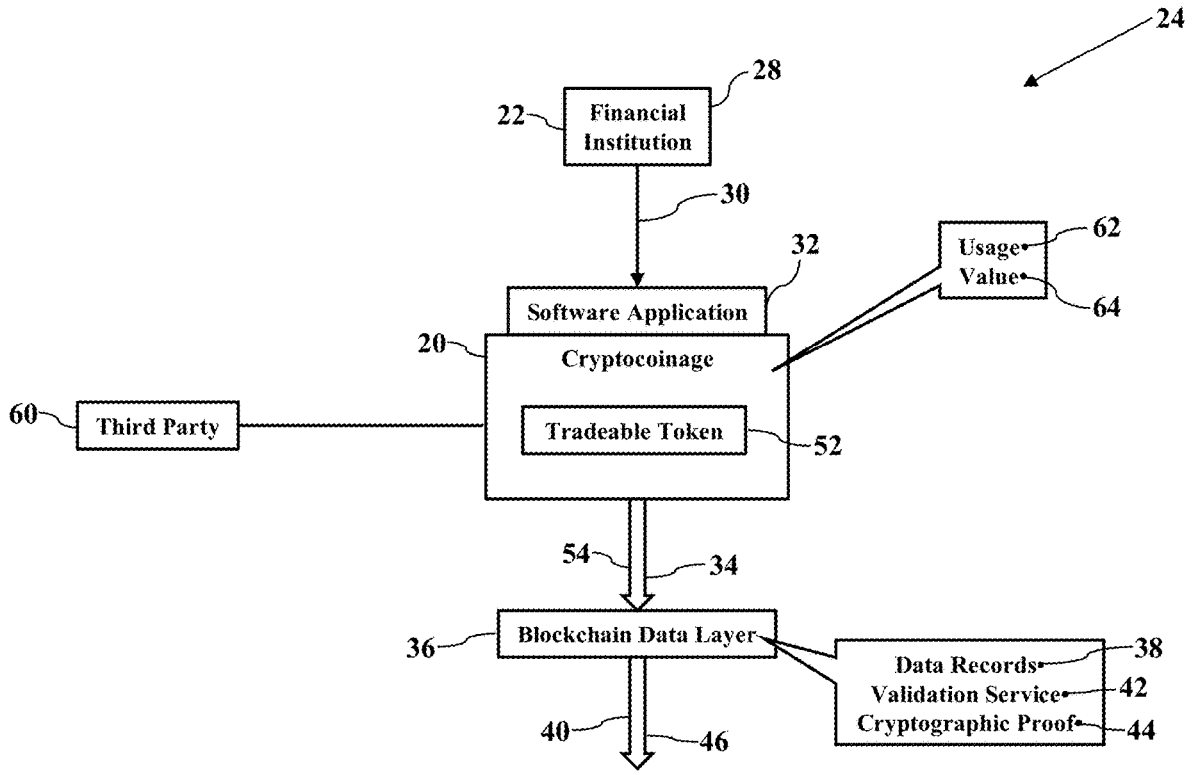


FIG. 1

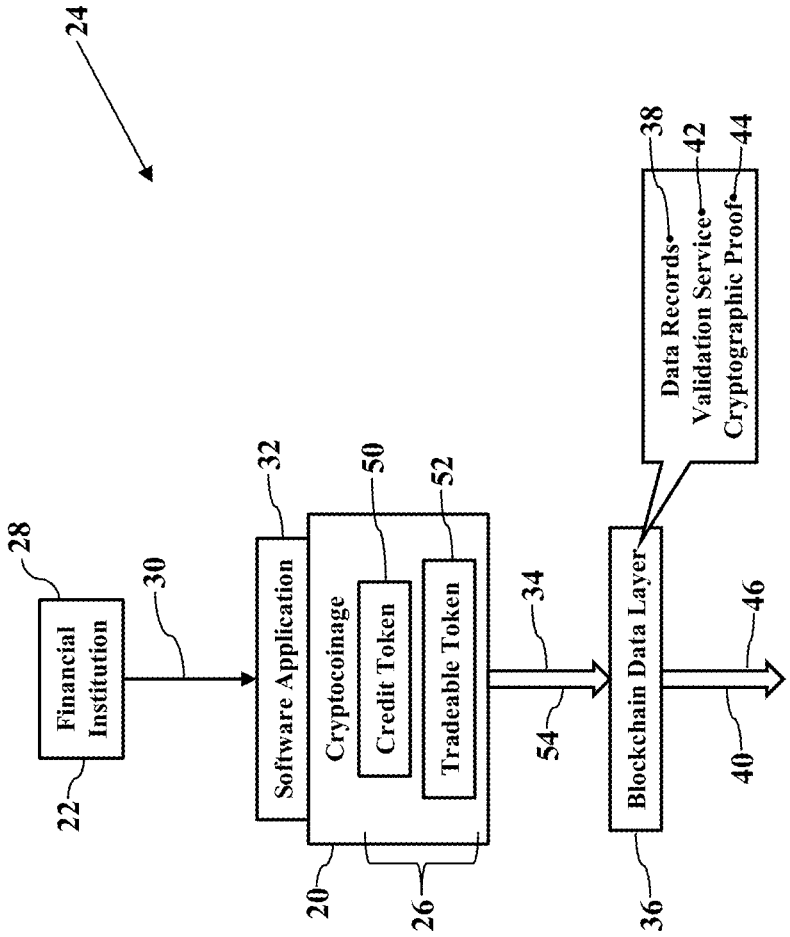

24
28
Financial Institution
22
30
32
Software Application
Cryptocoinage
50
Credit Token
52
Tradeable Token
20
26
34
54
Blockchain Data Layer
36
38
Data Records
42
Validation Service
44
Cryptographic Proof
46
40

FIG. 2

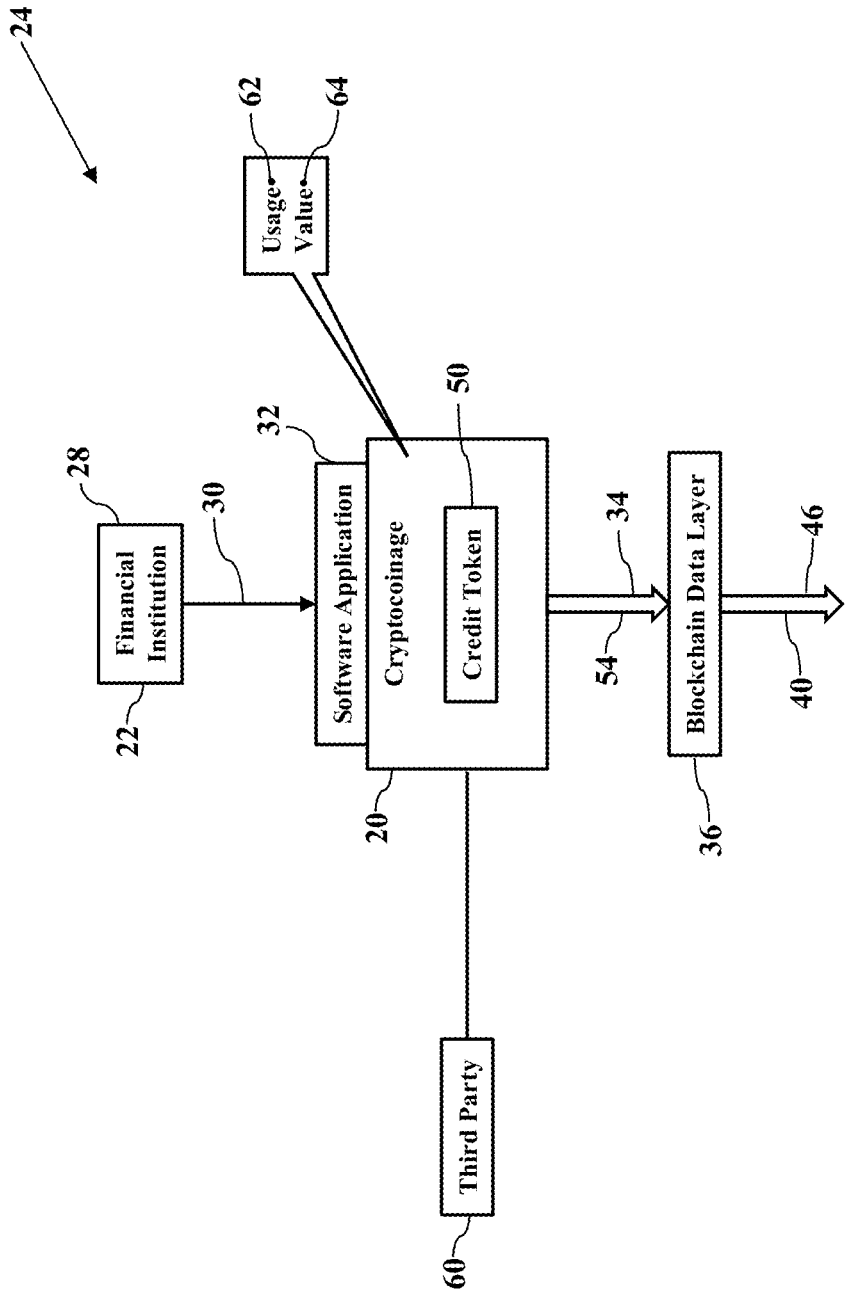

24
62
64
Usage
Value
28
Financial Institution
22
30
32
Software Application
Cryptocoinage
50
Credit Token
20
34
54
Blockchain Data Layer
36
46
40
Third Party
60

FIG. 3

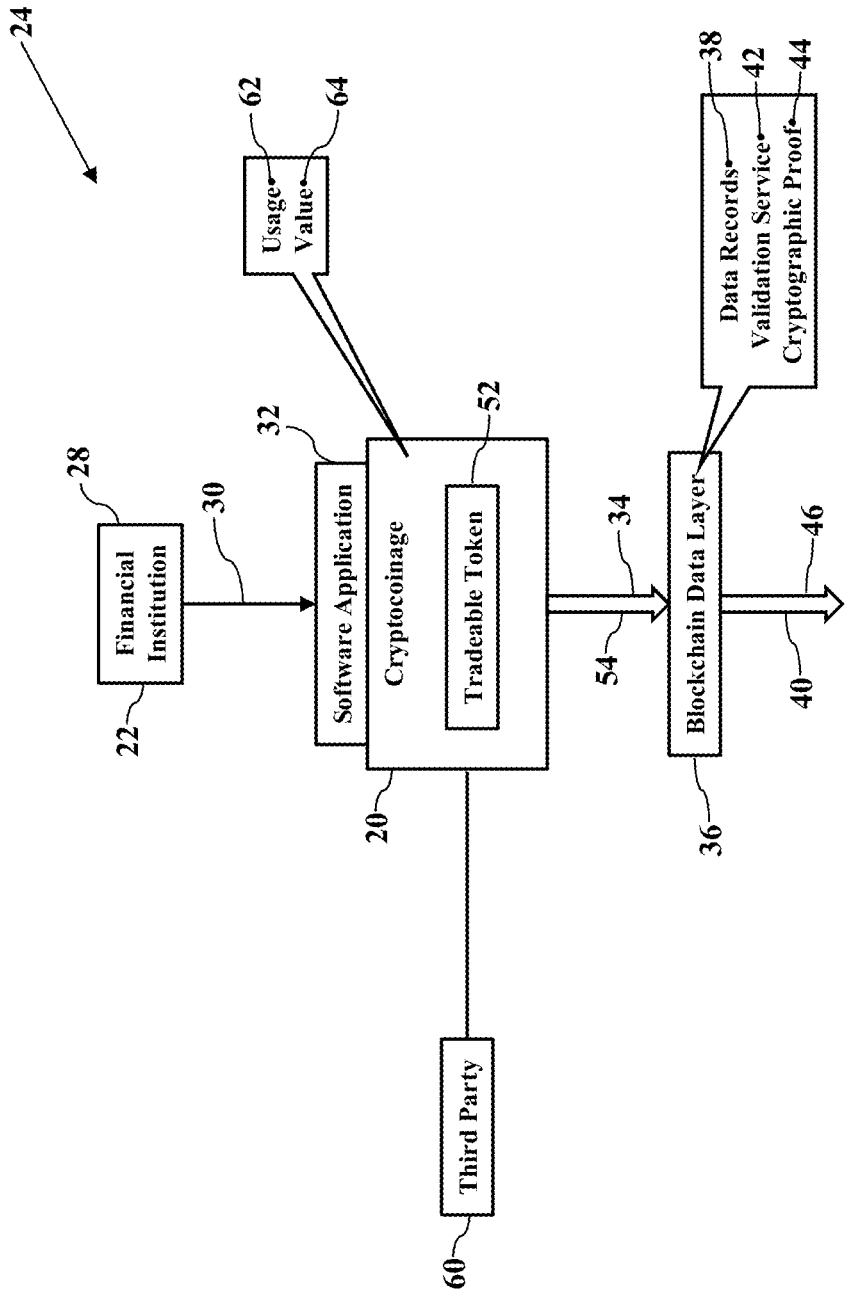

24
62
64
Usage
Value
38
42
44
Data Records
Validation Service
Cryptographic Proof
32
52
28
30
Financial Institution
Software Application
Cryptocoinage
Tradeable Token
34
46
54
Blockchain Data Layer
40
22
20
36
Third Party
60

FIG. 4

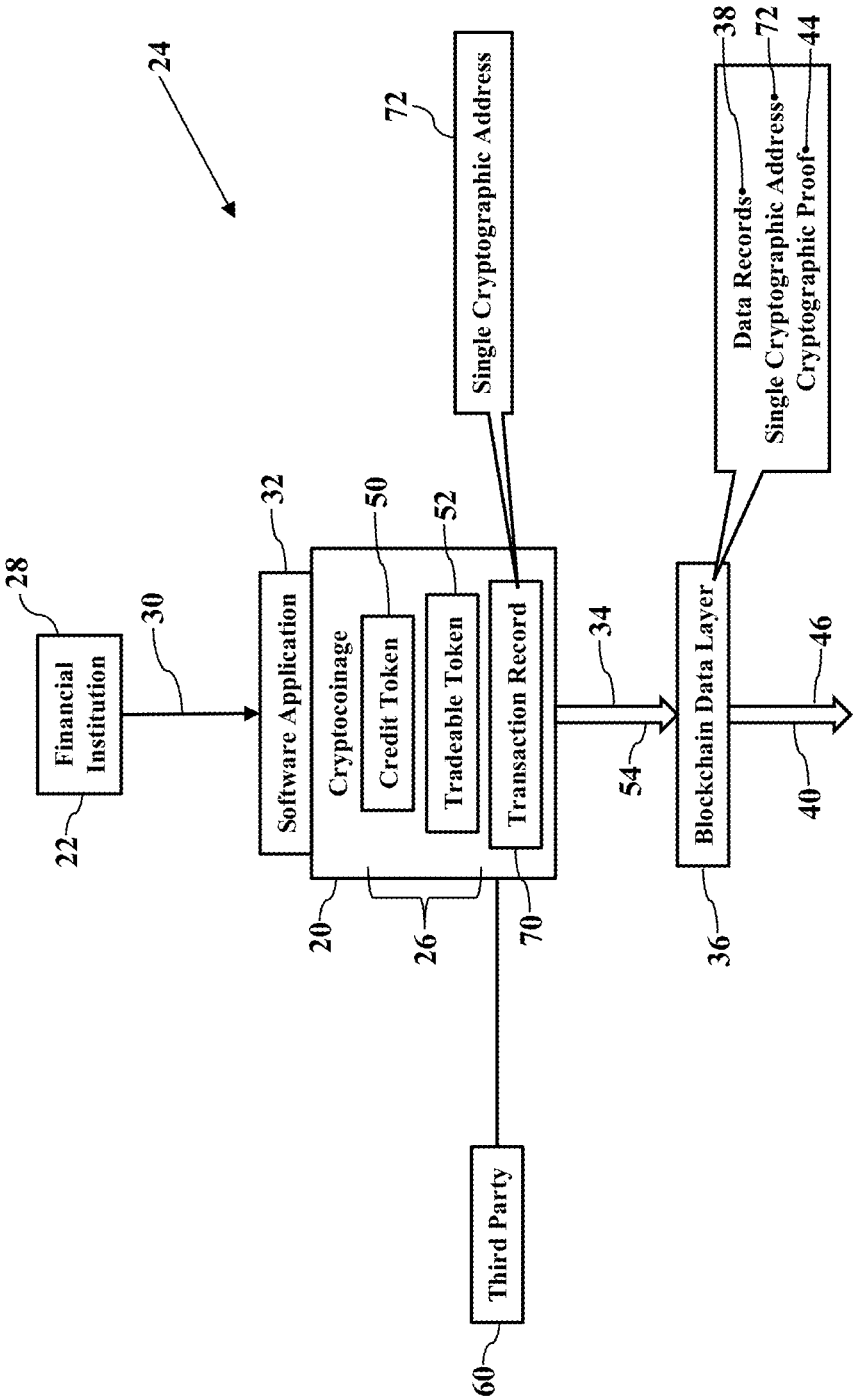

24
72
Single Cryptographic Address
38
72
44
Data Records
Single Cryptographic Address
Cryptographic Proof
28
Financial Institution
22
30
32
Software Application
Cryptocoinage
50
Credit Token
52
Tradeable Token
Transaction Record
20
26
70
34
54
Blockchain Data Layer
36
46
40
Third Party
60

FIG. 5

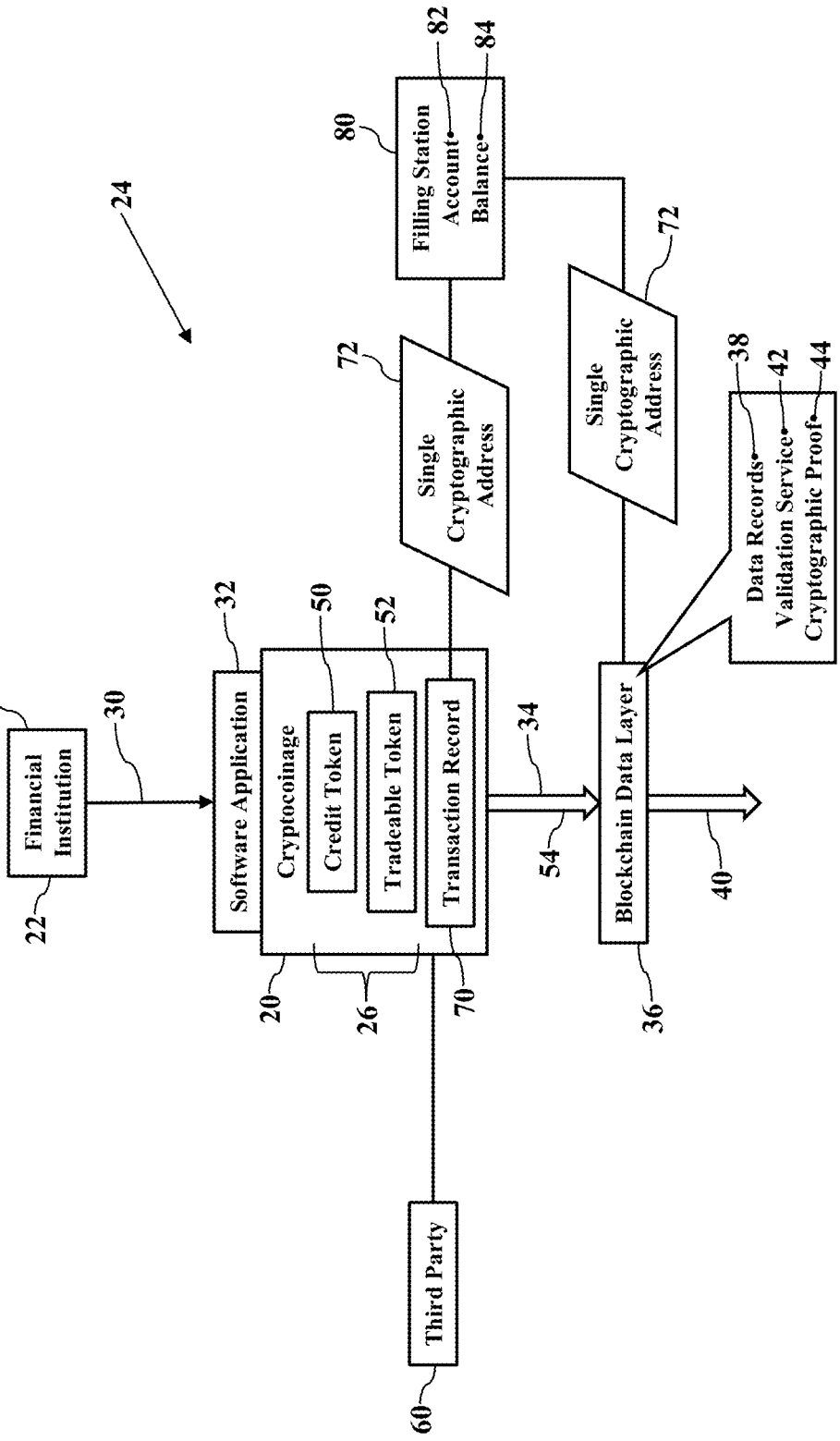

24
28
Financial Institution
22
30
32
Software Application
Cryptocoinage
50
Credit Token
52
Tradeable Token
Transaction Record
20
26
70
34
54
Blockchain Data Layer
36
40
72
Single Cryptographic Address
72
Single Cryptographic Address
80
Filling Station
82
Account
84
Balance
38
Data Records
42
Validation Service
44
Cryptographic Proof
60
Third Party

FIG. 6

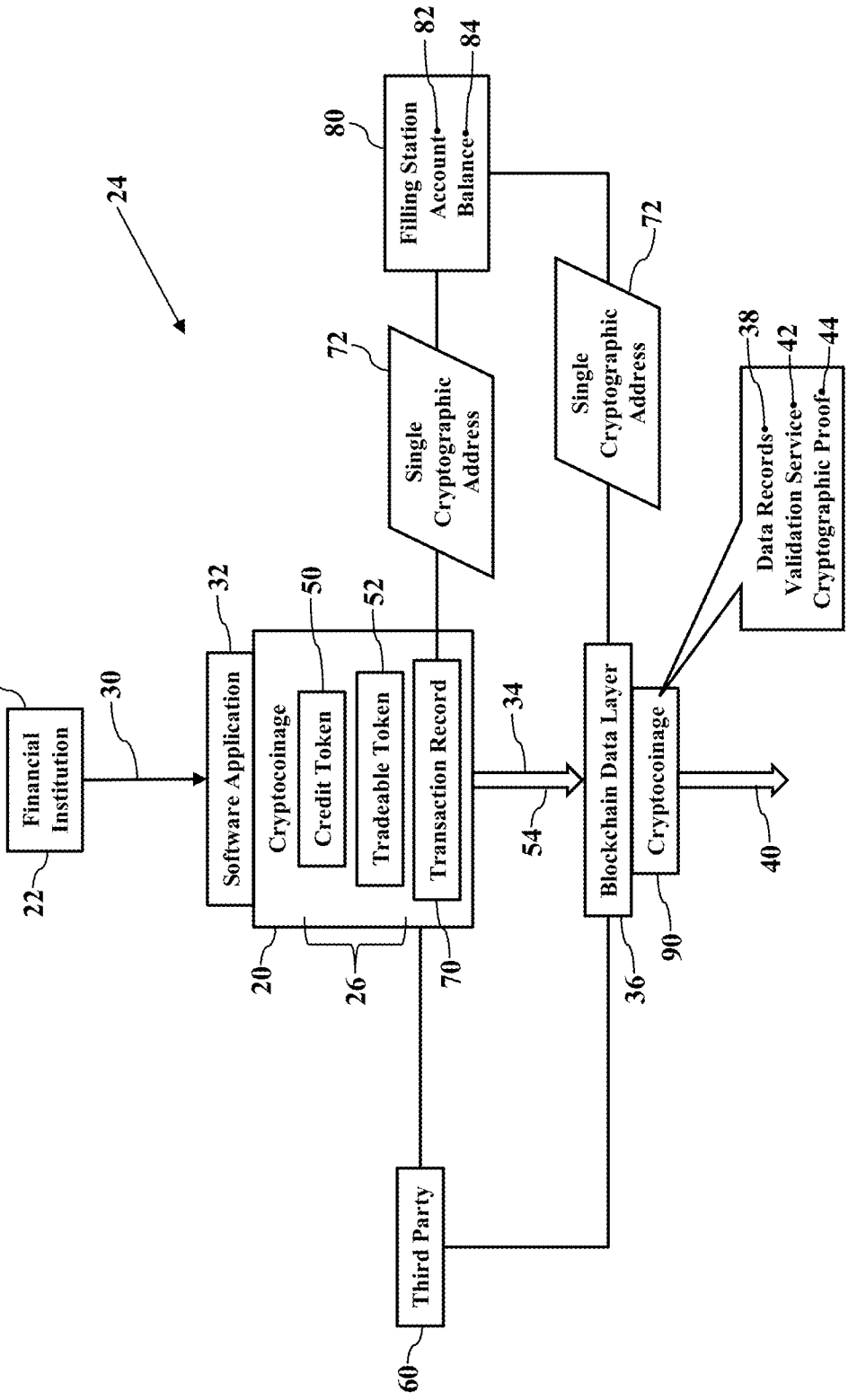

24
Financial Institution
28
22
30
Software Application
32
Cryptocoinage
Credit Token
50
Tradeable Token
52
Transaction Record
34
20
26
70
Third Party
60
Filling Station
80
Account
82
Balance
84
Single Cryptographic Address
72
Single Cryptographic Address
72
Blockchain Data Layer
54
Cryptocoinage
36
90
40
Data Records
38
Validation Service
42
Cryptographic Proof
44

FIG. 7

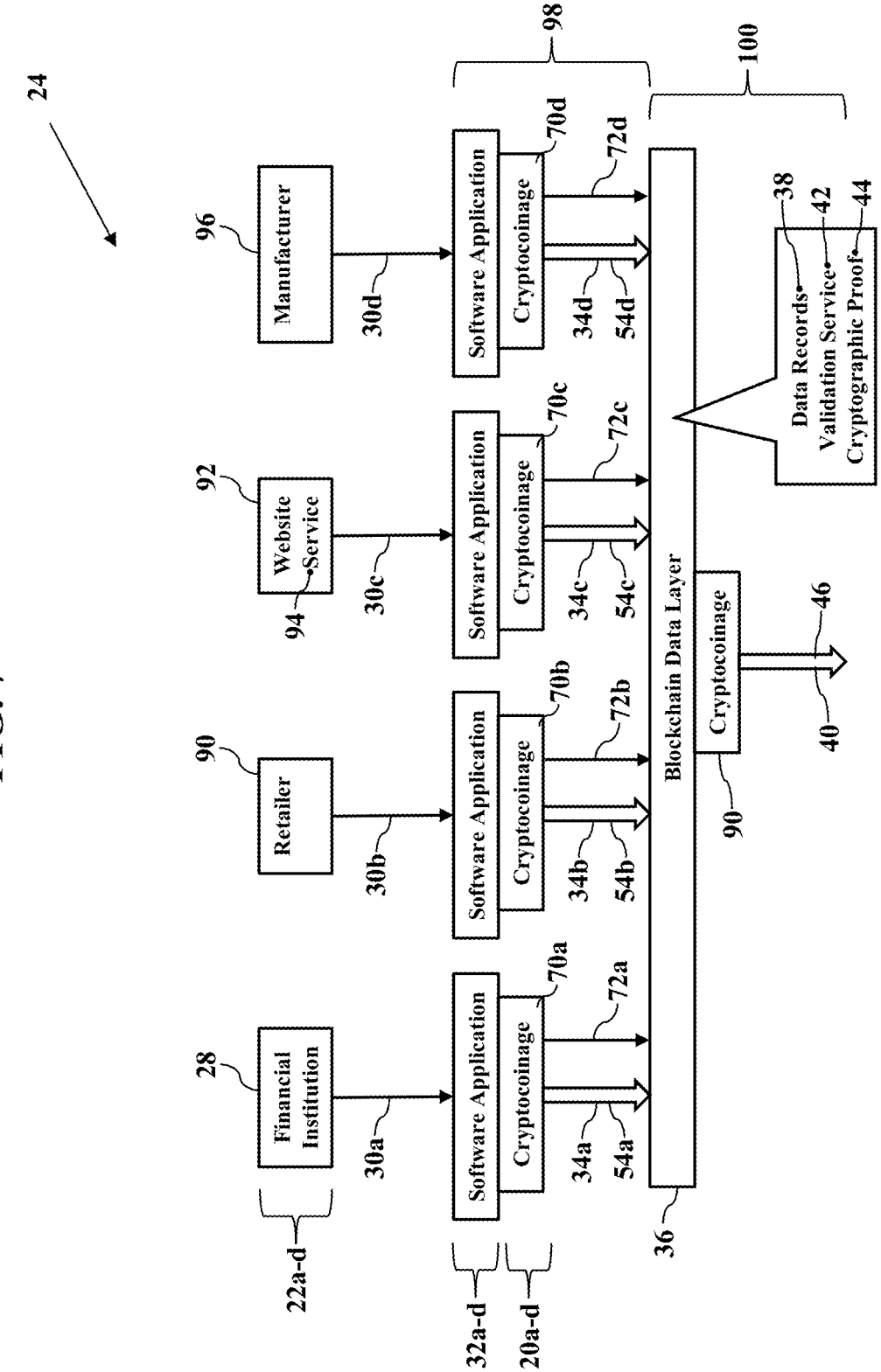

24
28
Financial Institution
22a-d
30a
Software Application
Cryptocoinage
70a
72a
34a
54a
32a-d
20a-d
90
Retailer
30b
Software Application
Cryptocoinage
70b
72b
34b
54b
92
Website Service
94
30c
Software Application
Cryptocoinage
70c
72c
34c
54c
96
Manufacturer
30d
Software Application
Cryptocoinage
70d
72d
34d
54d
98
100
Blockchain Data Layer
36
Cryptocoinage
90
40
46
38
Data Records
42
Validation Service
44
Cryptographic Proof

FIG. 8

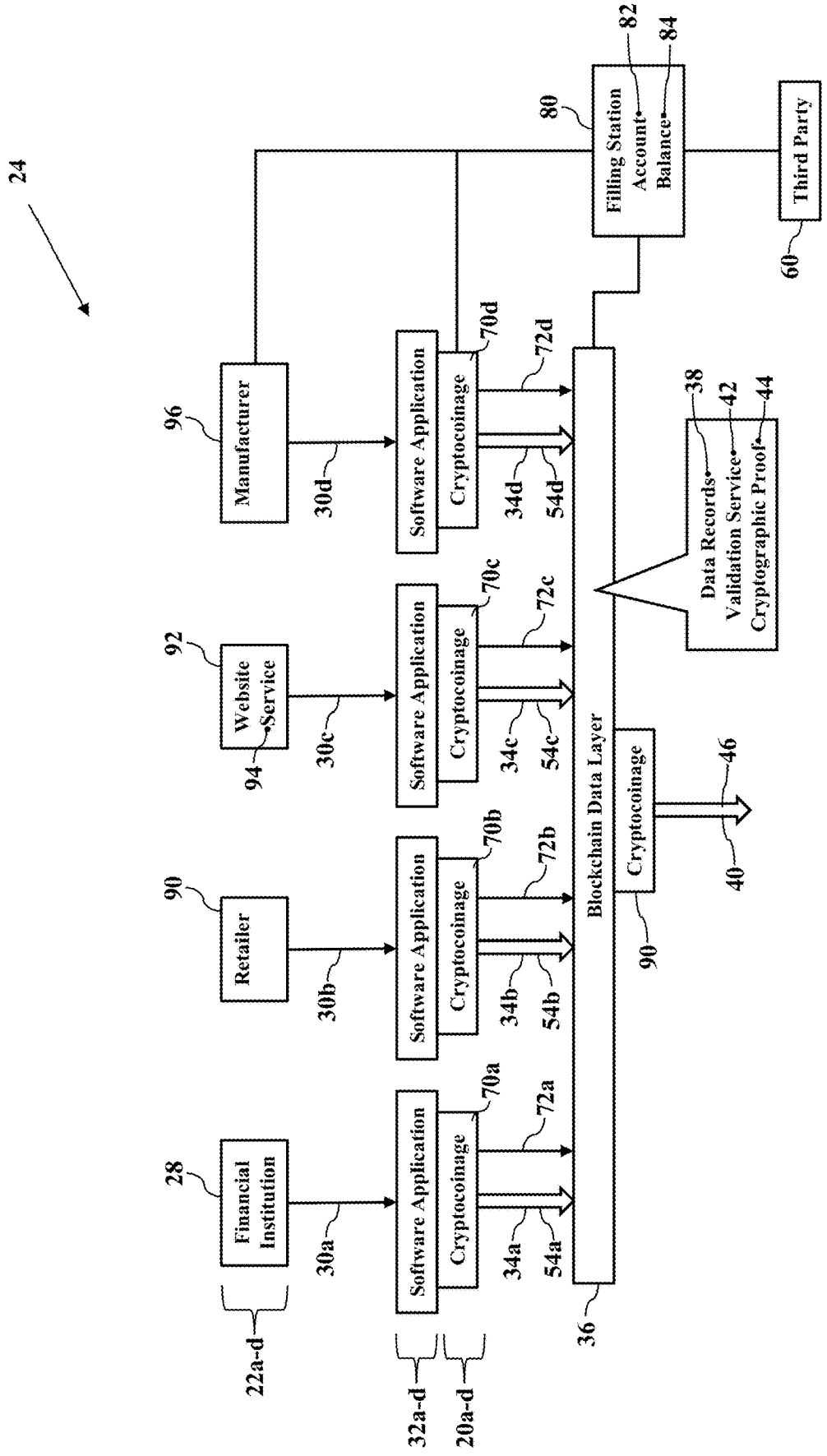

24
28
Financial Institution
22a-d
30a
Software Application
Cryptocoinage
70a
72a
34a
54a
32a-d
20a-d
90
Retailer
30b
Software Application
Cryptocoinage
70b
72b
34b
54b
92
Website Service
94
30c
Software Application
Cryptocoinage
70c
72c
34c
54c
96
Manufacturer
30d
Software Application
Cryptocoinage
70d
72d
34d
54d
36
Blockchain Data Layer
Cryptocoinage
90
40
46
38
Data Records
42
Validation Service
44
Cryptographic Proof
80
Filling Station
82
Account
84
Balance
Third Party
60

FIG. 9

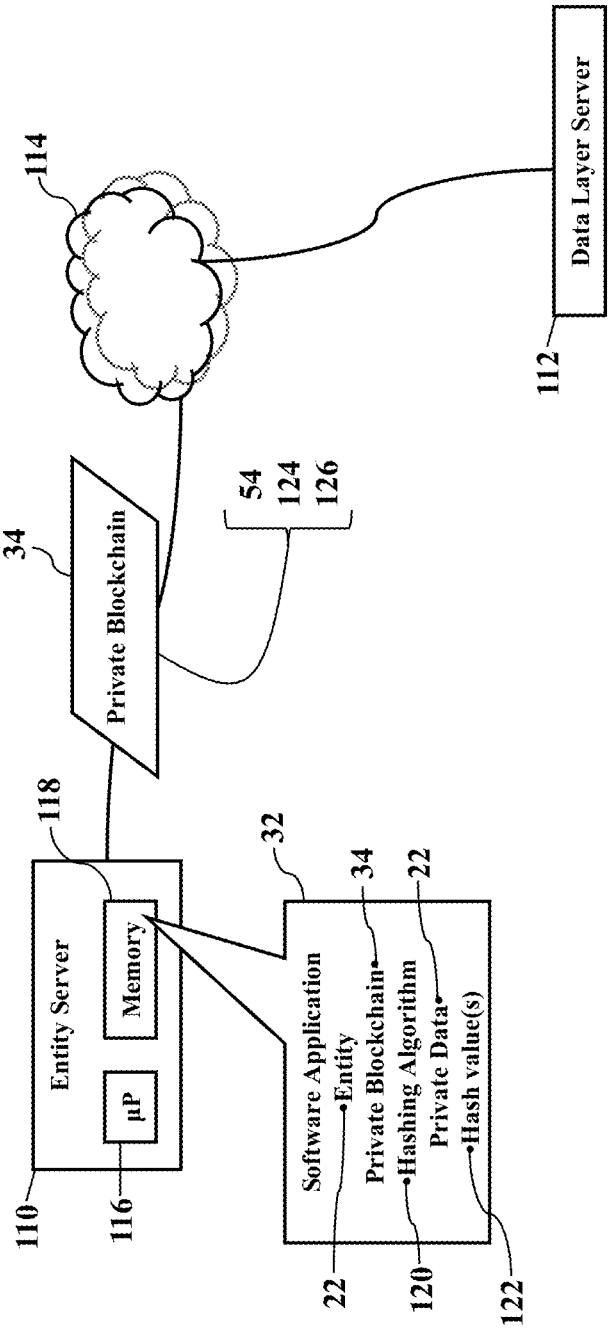

Data Layer Server
112
114
Private Blockchain
34
54
124
126
118
32
34
22
Entity Server
Memory
µP
Software Application
Entity
Private Blockchain
Hashing Algorithm
Private Data
Hash value(s)
110
116
22
120
122

FIG. 10

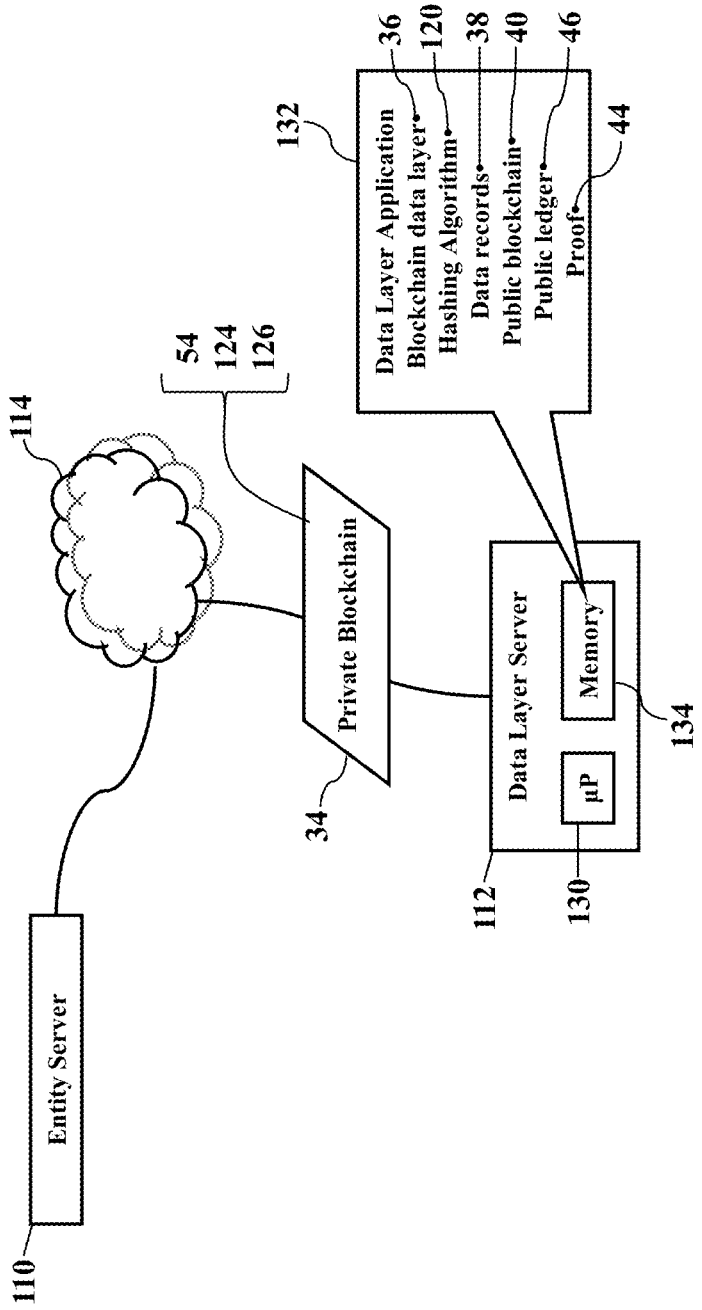

Entity Server
110
114
54
124
126
Private Blockchain
34
Data Layer Application
Blockchain data layer
Hashing Algorithm
Data records
Public blockchain
Public ledger
Proof
36
120
38
40
46
44
132
Data Layer Server
μP
Memory
112
130
134

FIG. 11

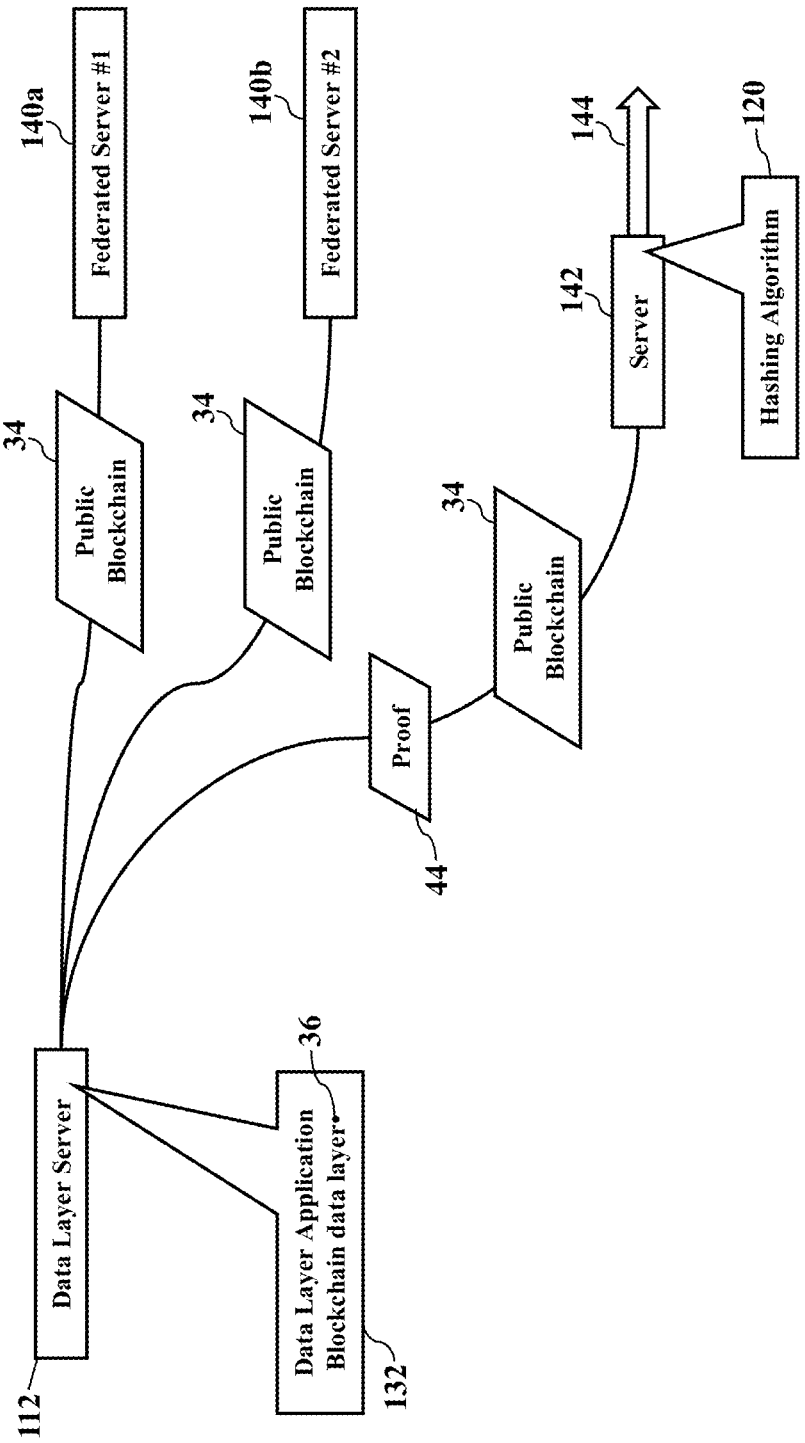

Federated Server #1
140a
Federated Server #2
140b
144
120
142
Server
Hashing Algorithm
34
Public Blockchain
34
Public Blockchain
34
Public Blockchain
Proof
44
Data Layer Server
112
Data Layer Application Blockchain data layer
36
132

FIG. 12

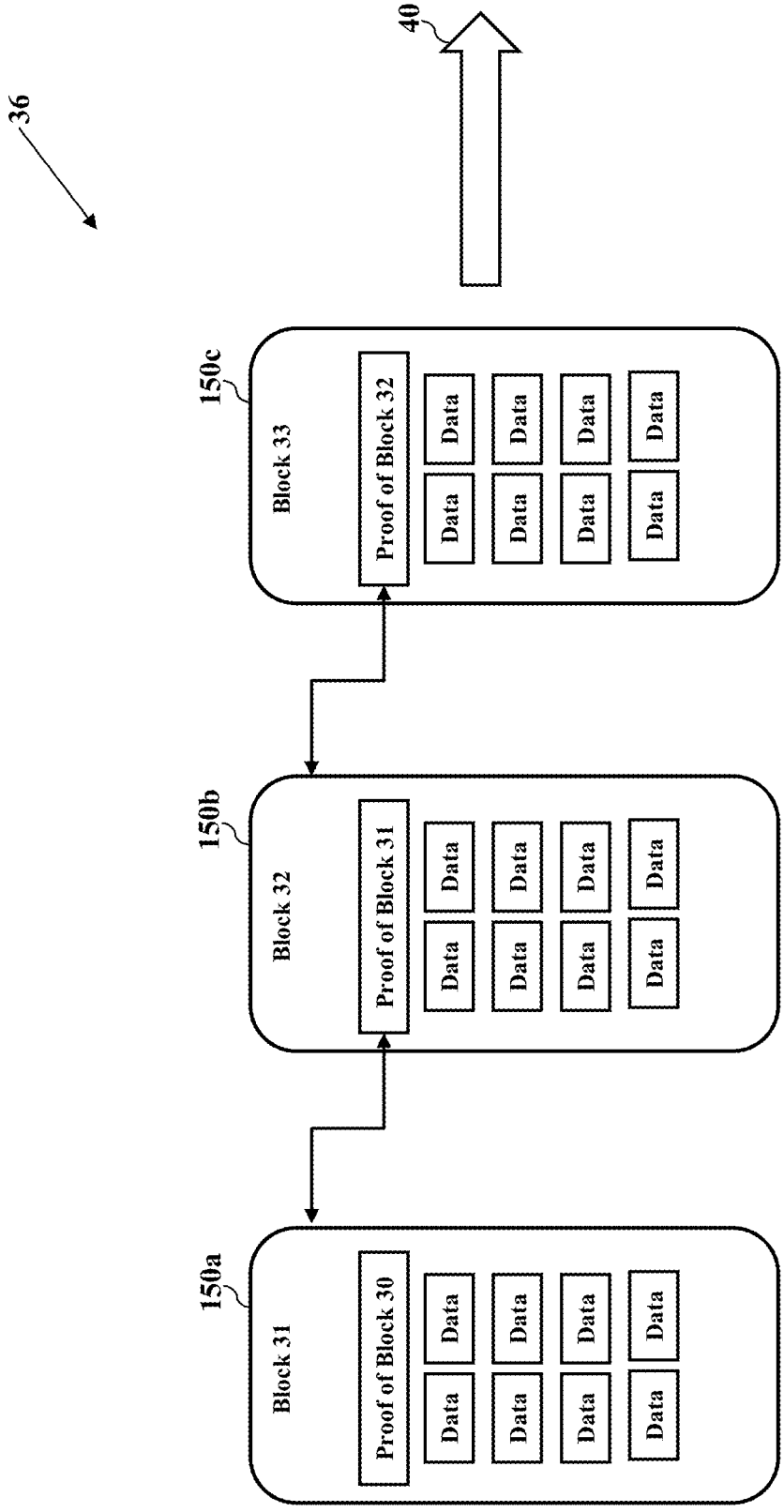

36
40
150a
Block 31
Proof of Block 30
Data
150b
Block 32
Proof of Block 31
Data
150c
Block 33
Proof of Block 32
Data

FIG. 13

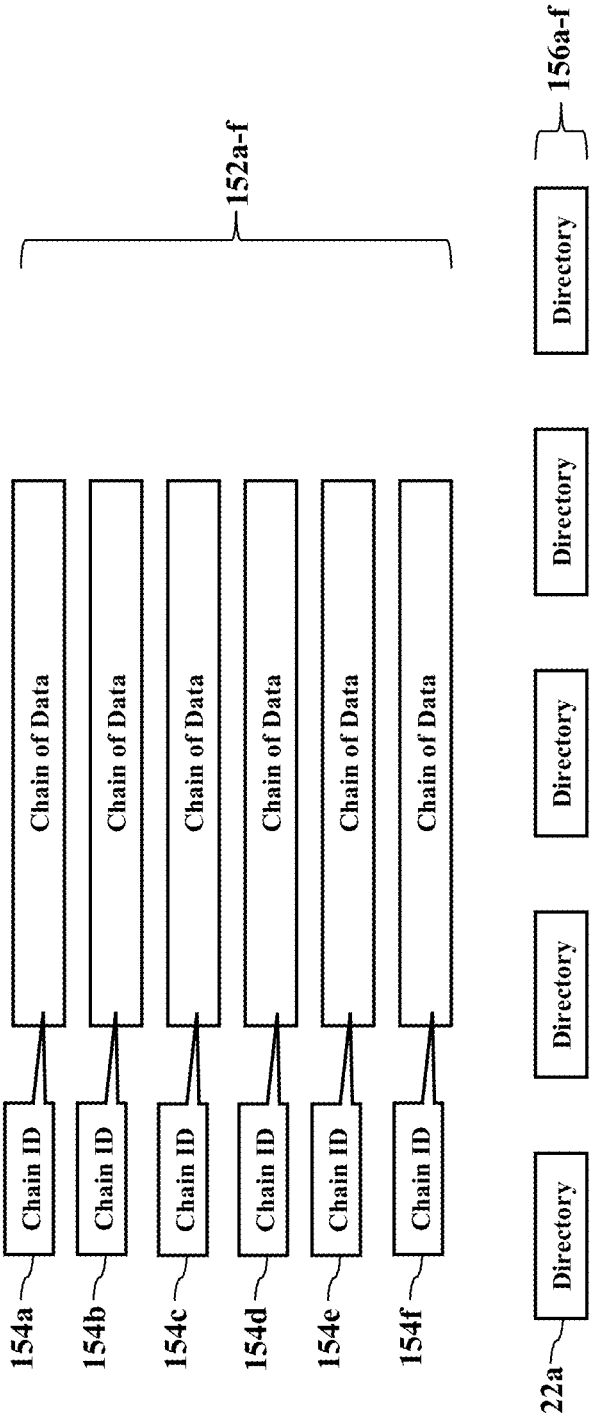

152a-f
156a-f
Directory
Directory
Directory
Directory
Directory
Directory
Chain of Data
Chain of Data
Chain of Data
Chain of Data
Chain of Data
Chain of Data
Chain ID
Chain ID
Chain ID
Chain ID
Chain ID
Chain ID
154a
154b
154c
154d
154e
154f
22a

FIG. 14

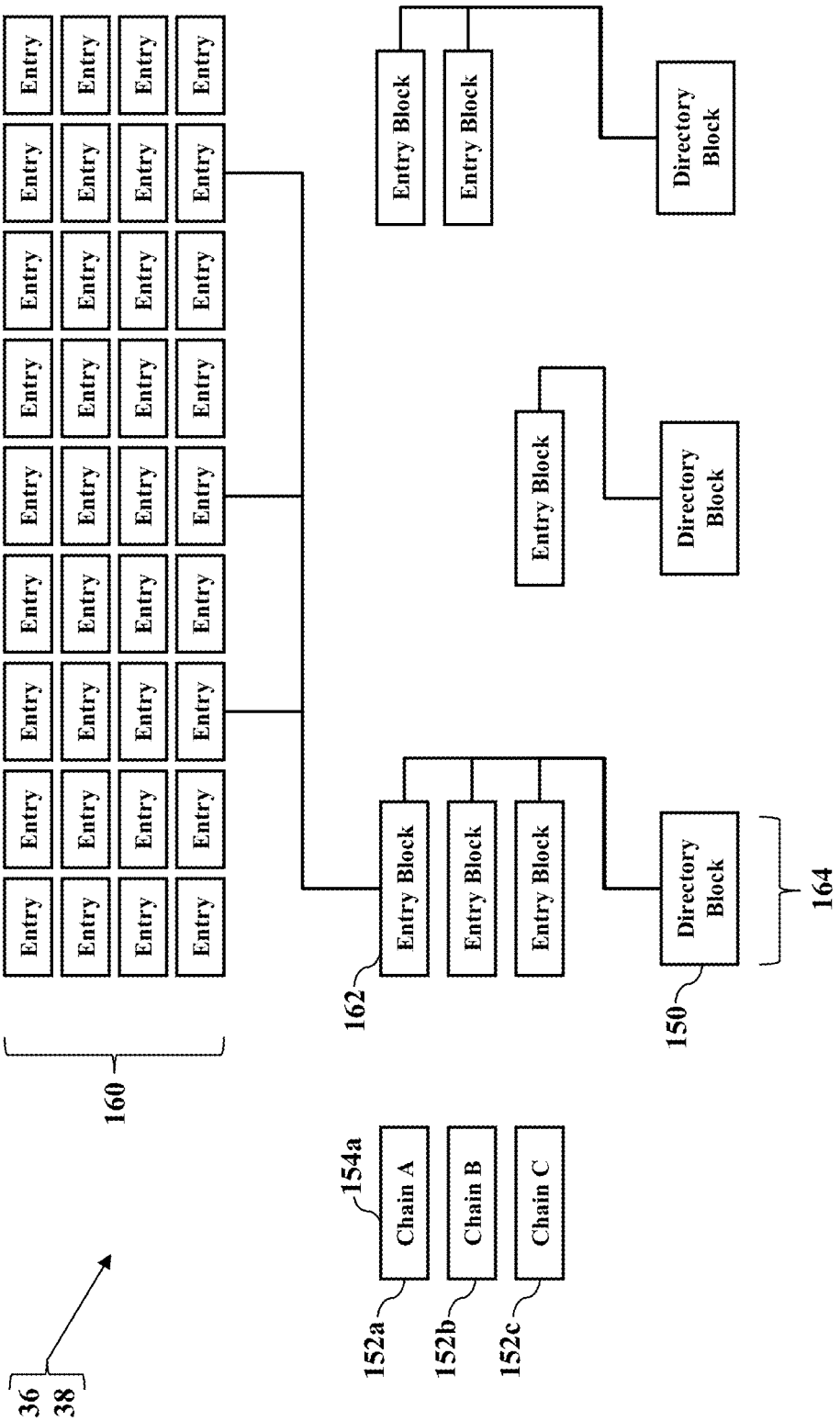

Entry
Entry Block
Directory Block
160
162
150
164
154a
Chain A
Chain B
Chain C
152a
152b
152c
36
38

FIG. 15

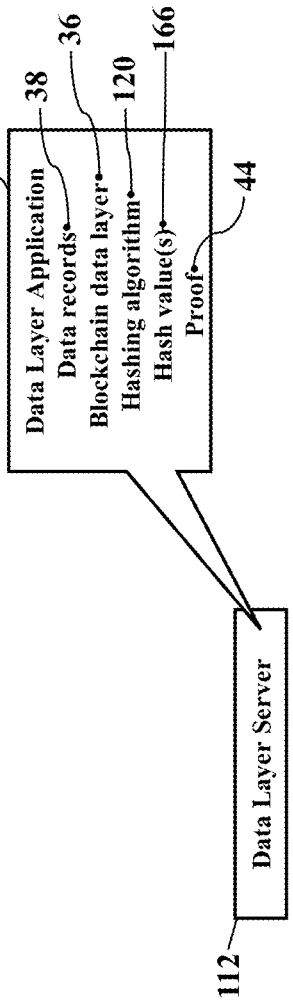

132
Data Layer Application
38
Data records
36
Blockchain data layer
120
Hashing algorithm
166
Hash value(s)
44
Proof
Data Layer Server
112

FIG. 16

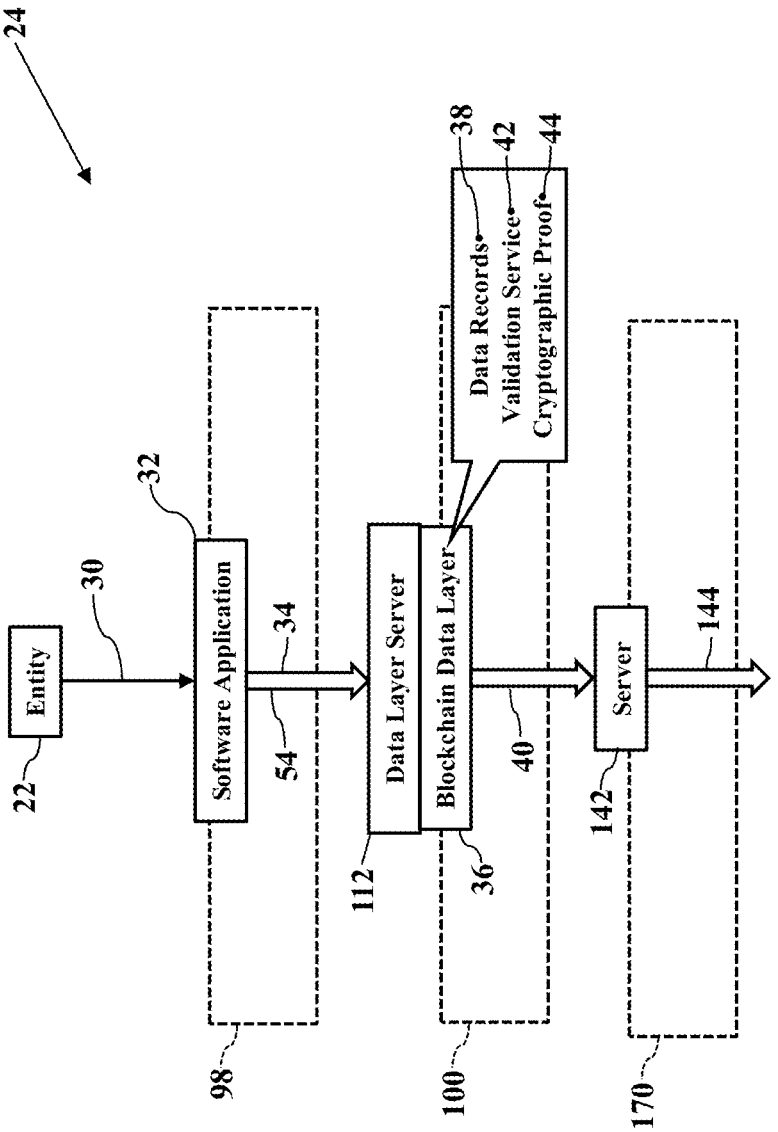

24
38
42
44
Data Records
Validation Service
Cryptographic Proof
32
30
34
Entity
Software Application
Data Layer Server
Blockchain Data Layer
Server
144
22
54
112
36
40
142
98
100
170

FIG. 17

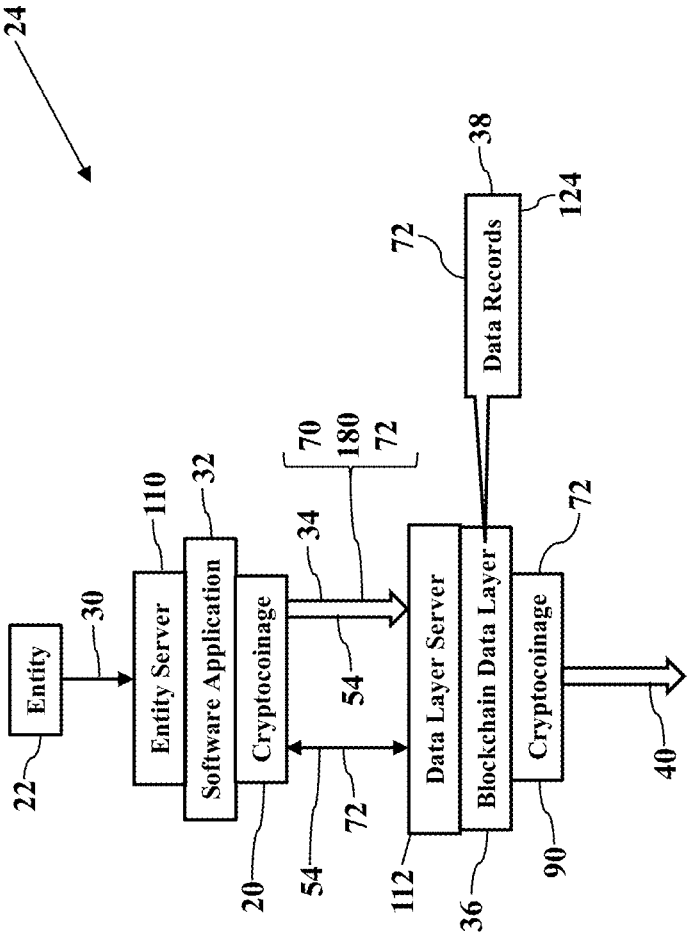

24
Entity
22
30
Entity Server
110
Software Application
32
Cryptocoinage
20
34
70
180
72
54
72
54
Data Layer Server
112
Blockchain Data Layer
36
72
Cryptocoinage
90
72
Data Records
124
38
40

FIG. 18

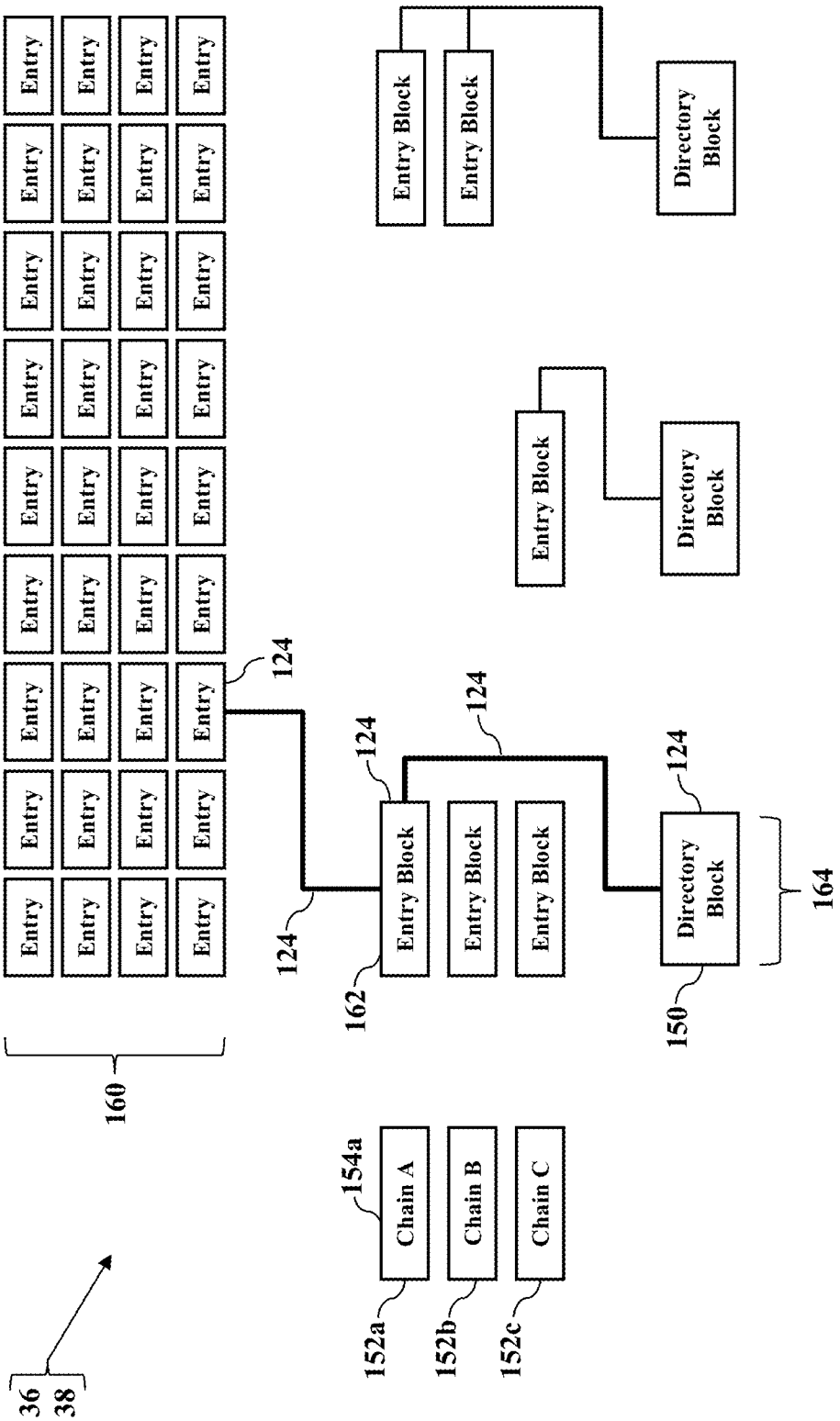

Entry
Entry Block
Directory Block
Chain A
Chain B
Chain C
124
154a
152a
152b
152c
160
162
150
164
36
38

FIG. 19

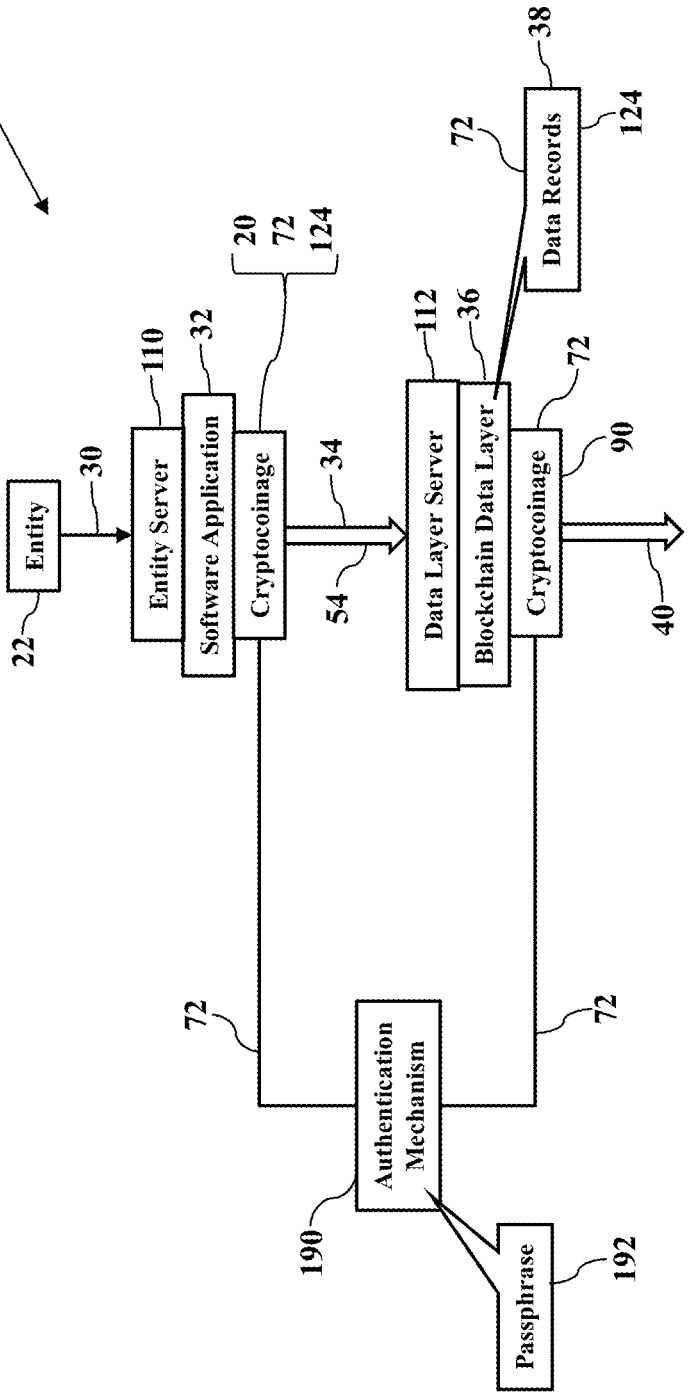

24
Entity
22
30
Entity Server
110
Software Application
32
Cryptocoinage
20
72
124
34
54
Data Layer Server
112
Blockchain Data Layer
36
Data Records
38
72
124
Cryptocoinage
72
90
40
72
72
Authentication Mechanism
190
Passphrase
192

FIG. 20

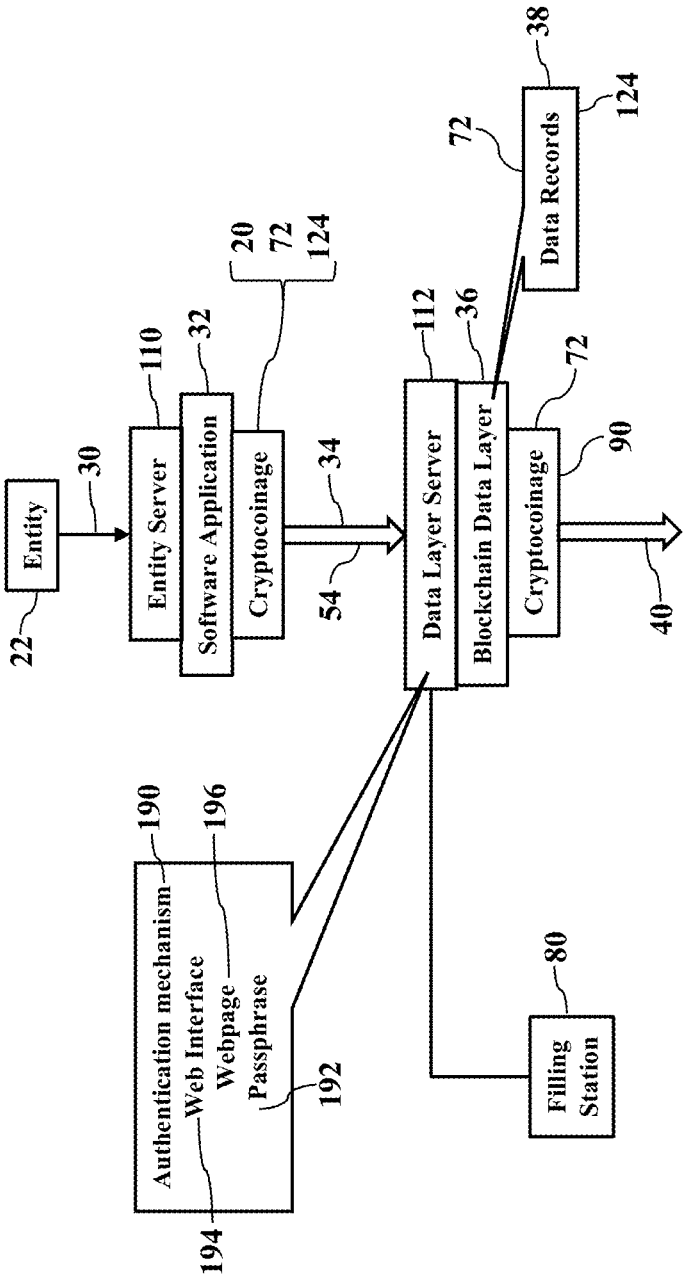

Entity
22
30
Entity Server
110
Software Application
32
Cryptocoinage
20
72
124
34
54
Data Layer Server
112
Blockchain Data Layer
36
Cryptocoinage
72
90
40
Data Records
38
72
124
Authentication mechanism
190
Web Interface
196
Webpage
Passphrase
192
194
Filling Station
80

FIG. 21

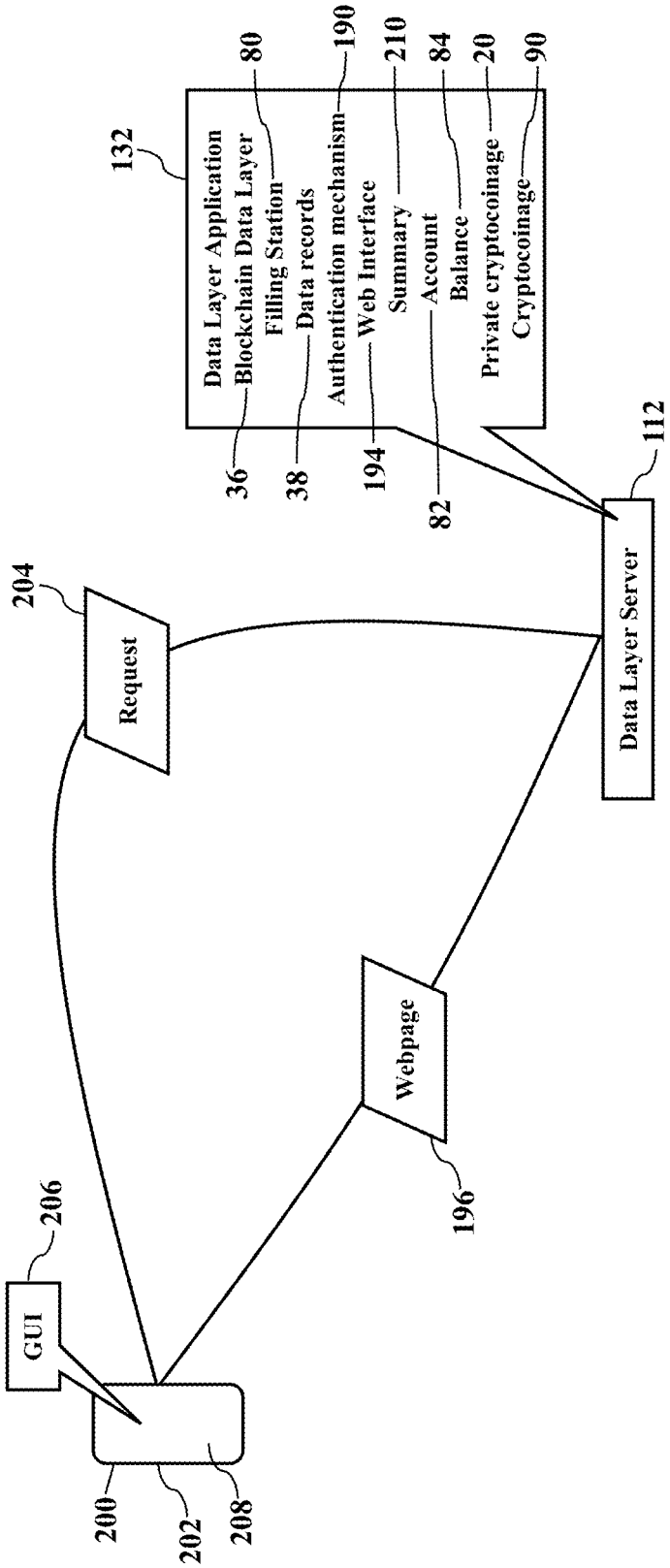

132
Data Layer Application
Blockchain Data Layer
Filling Station
Data records
Authentication mechanism
Web Interface
Summary
Account
Balance
Private cryptocoinage
Cryptocoinage
80
190
210
84
20
90
36
38
194
82
112
Data Layer Server
204
Request
206
GUI
Webpage
196
200
202
208

FIG. 22

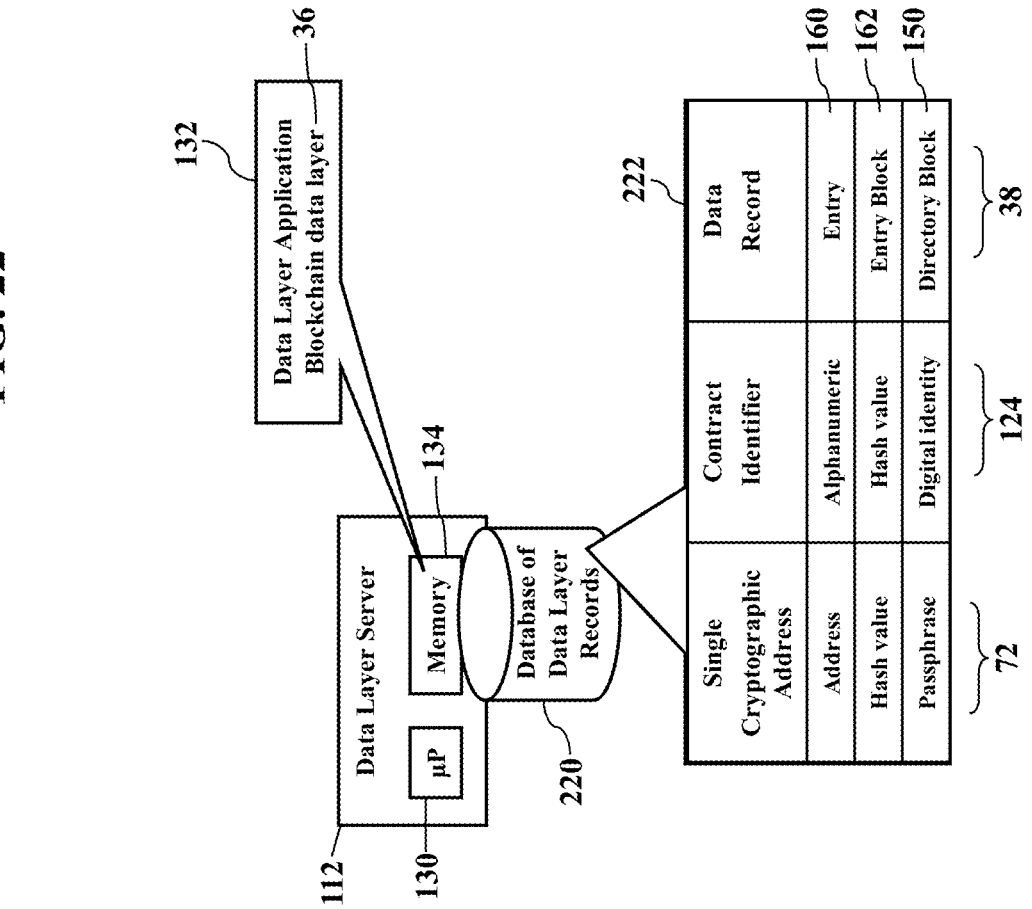

Data Layer Application Blockchain data layer
36
132
Data Layer Server
µP
Memory
134
112
130
Database of Data Layer Records
220
222
Single Cryptographic Address
Contract Identifier
Data Record
Address
Alphanumeric
Entry
160
Hash value
Hash value
Entry Block
162
Passphrase
Digital identity
Directory Block
150
72
124
38

FIG. 23

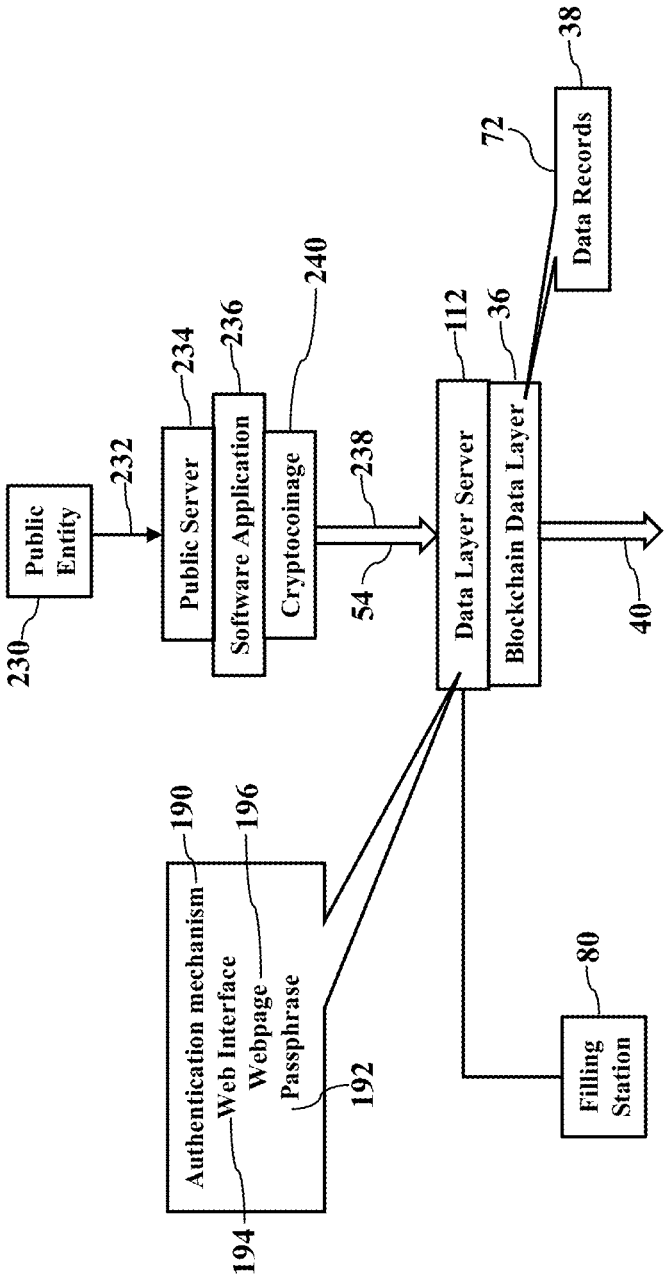

Public Entity
230
232
Public Server
234
Software Application
236
Cryptocoinage
240
238
54
Data Layer Server
112
Blockchain Data Layer
36
40
Data Records
72
38
Authentication mechanism
190
Web Interface
196
Webpage
Passphrase
192
194
Filling Station
80

FIG. 24

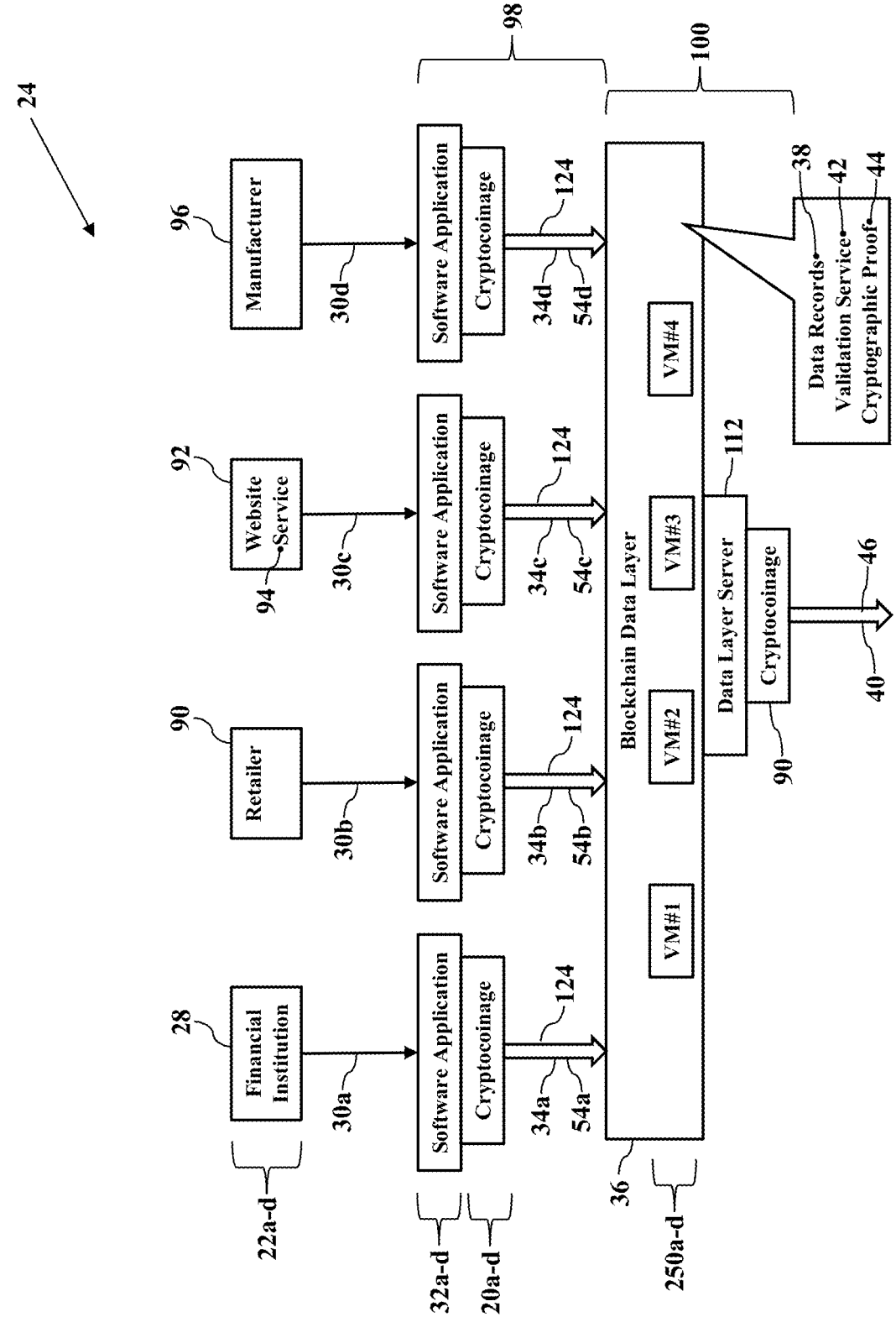

24
28
Financial Institution
22a-d
30a
Software Application
Cryptocoinage
32a-d
20a-d
34a
54a
124
90
Retailer
30b
34b
54b
92
Website Service
94
30c
34c
54c
96
Manufacturer
30d
34d
54d
98
100
36
250a-d
Blockchain Data Layer
VM#1
VM#2
VM#3
VM#4
112
Data Layer Server
90
40
46
38
Data Records
42
Validation Service
44
Cryptographic Proof

FIG. 25

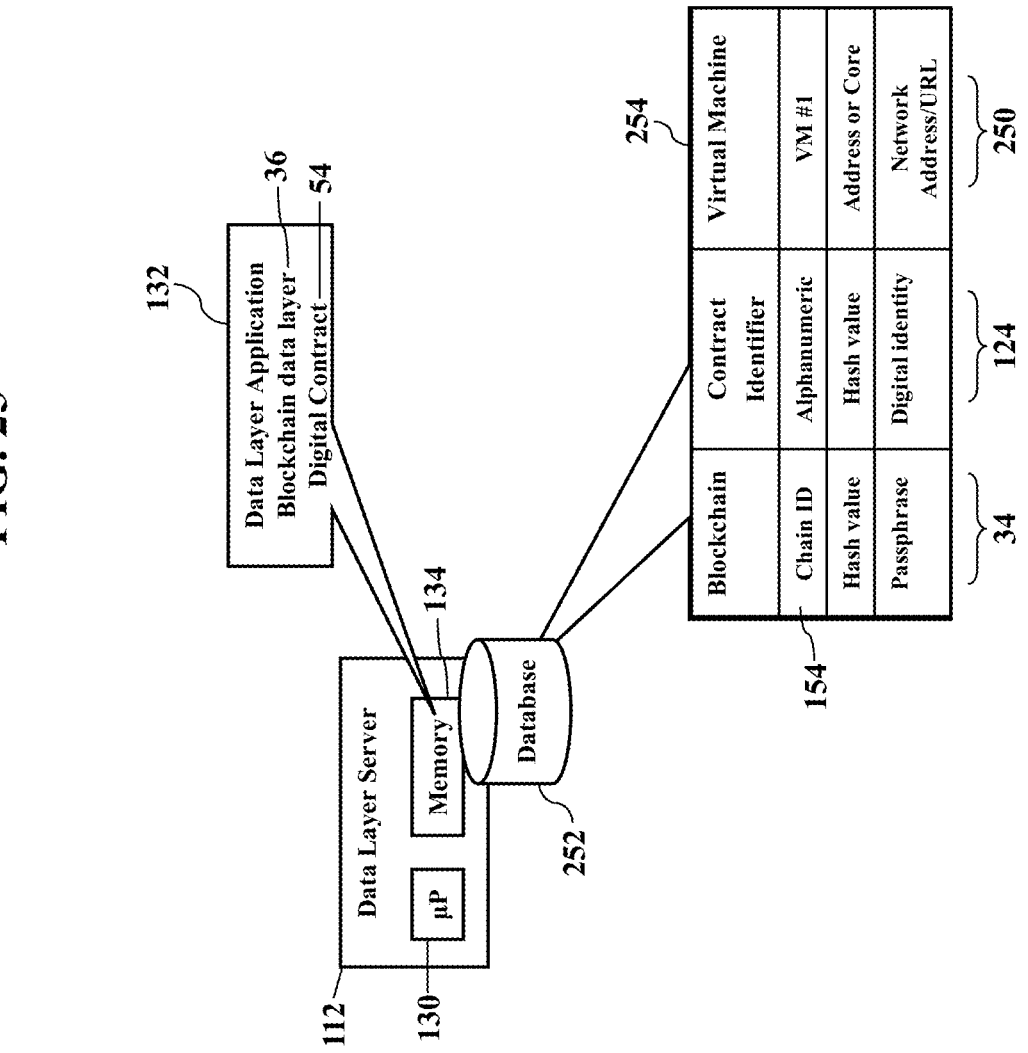

Data Layer Application
Blockchain data layer
Digital Contract
132
36
54
Data Layer Server
µP
Memory
134
Database
112
130
252
254
Blockchain
Contract Identifier
Virtual Machine
Chain ID
Alphanumeric
VM #1
Hash value
Hash value
Address or Core
Passphrase
Digital identity
Network Address/URL
154
34
124
250

FIG. 26

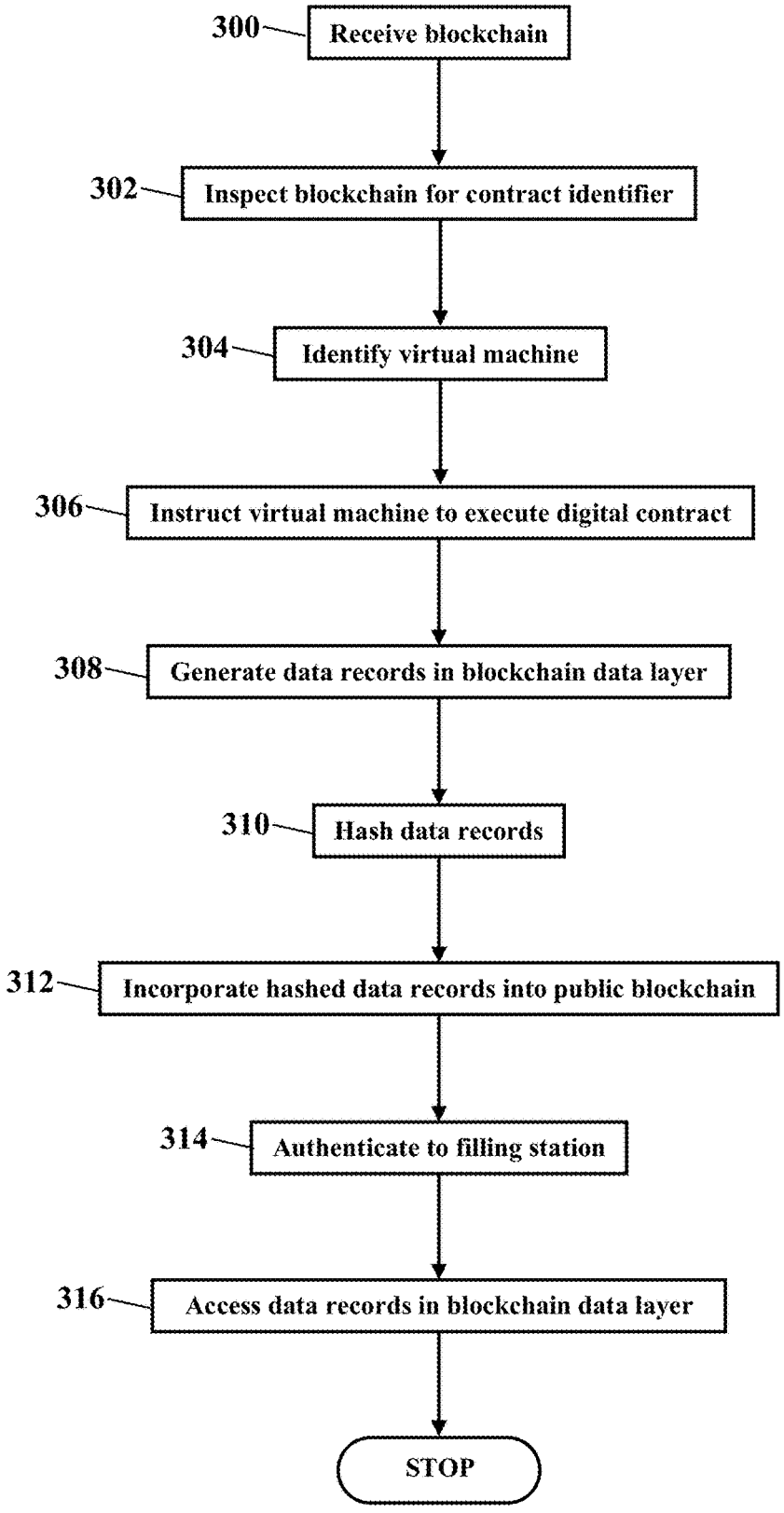

300
Receive blockchain
302
Inspect blockchain for contract identifier
304
Identify virtual machine
306
Instruct virtual machine to execute digital contract
308
Generate data records in blockchain data layer
310
Hash data records
312
Incorporate hashed data records into public blockchain
314
Authenticate to filling station
316
Access data records in blockchain data layer
STOP

FIG. 27

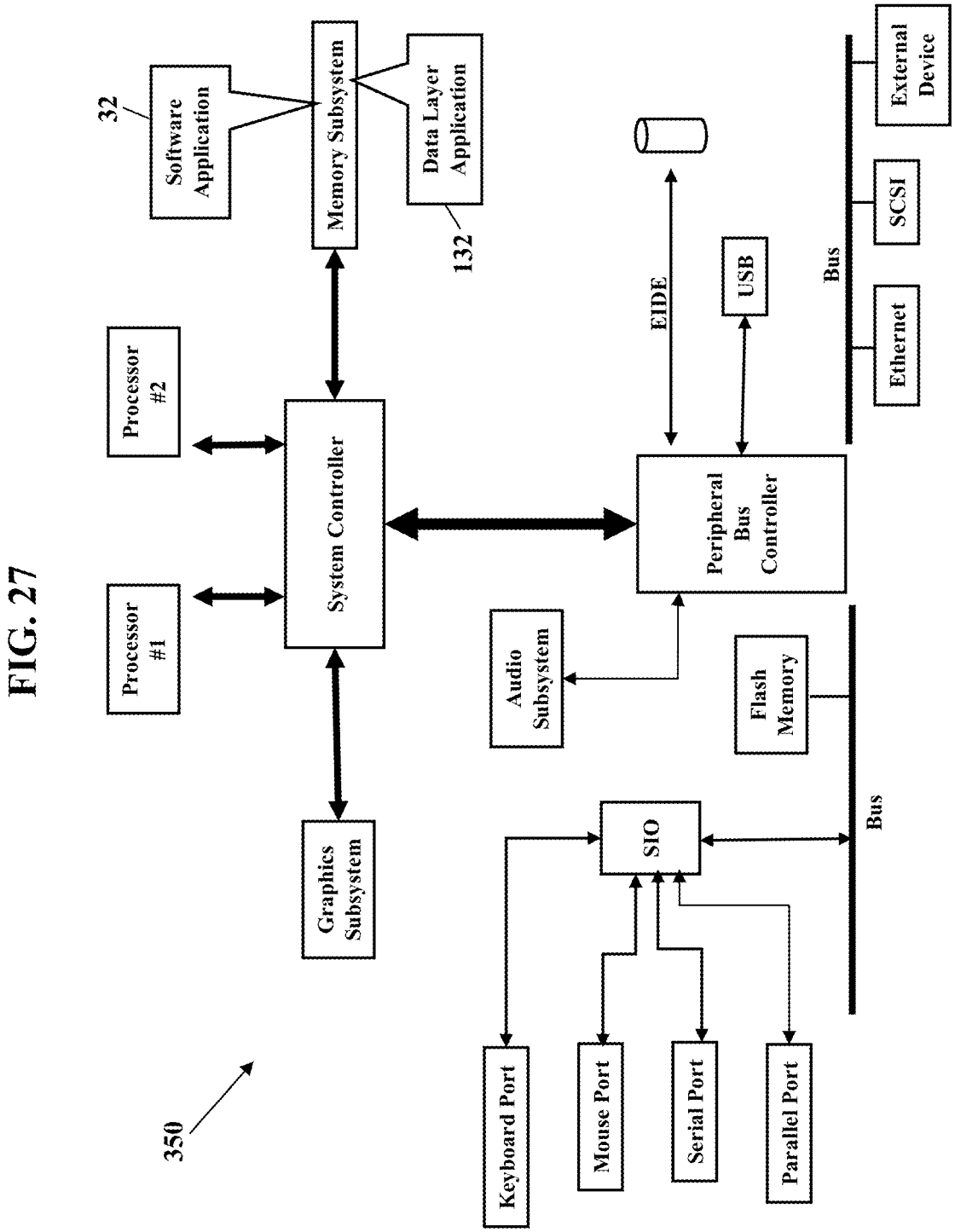
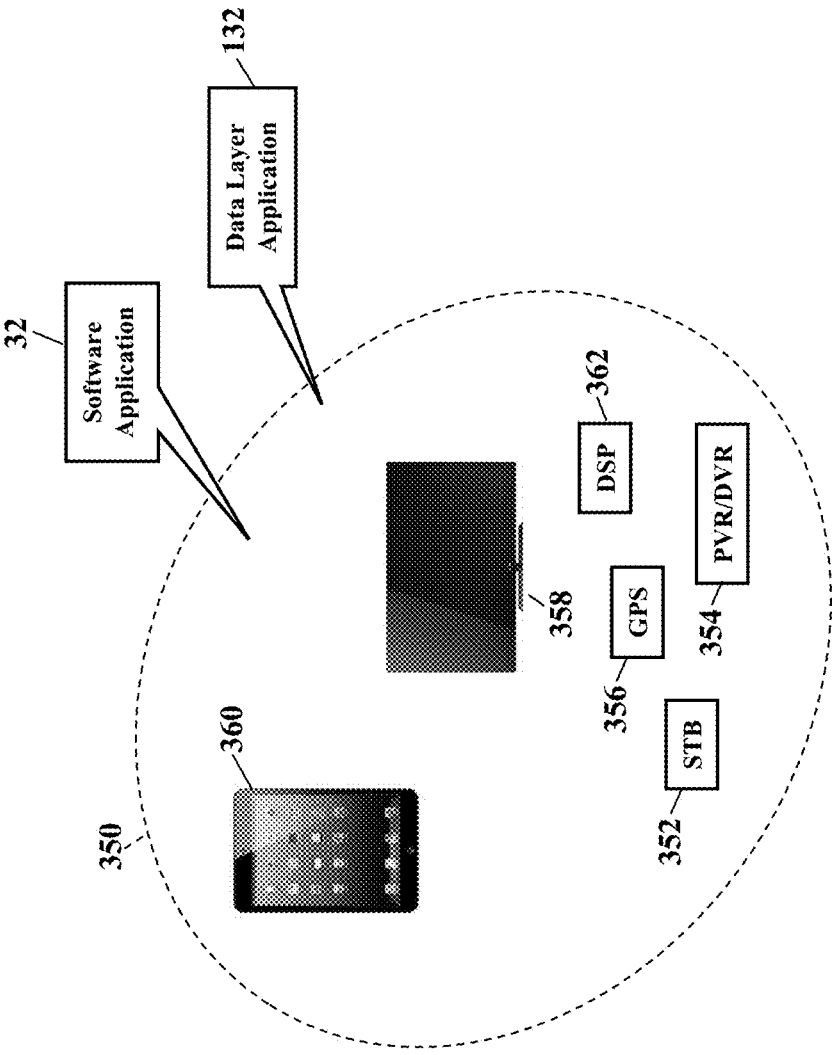

350
32
Software Application
Memory Subsystem
Data Layer Application
132
Processor #2
Processor #1
System Controller
Graphics Subsystem
Peripheral Bus Controller
EIDE
USB
Bus
External Device
SCSI
Ethernet
Audio Subsystem
Flash Memory
SIO
Bus
Keyboard Port
Mouse Port
Serial Port
Parallel Port

FIG. 28


132
Data Layer Application
32
Software Application
362
DSP
358
GPS
356
PVR/DVR
354
360
STB
352
350

DIGITAL CONTRACTS IN BLOCKCHAIN ENVIRONMENTS

CROSS-REFERENCE TO RELATED APPLICATIONS

[0001] This application claims domestic benefit of U.S. Provisional Application No. 62/714,909 filed Aug. 6, 2018 and incorporated herein by reference in its entirety. This application relates to U.S. application Ser. No. 15/983,572 filed May 18, 2018 and incorporated herein by reference in its entirety. This application also relates to U.S. application Ser. No. 15/983,595 filed May 18, 2018 and incorporated herein by reference in its entirety. This application also relates to U.S. application Ser. No. 15/983,612 filed May 18, 2018 and incorporated herein by reference in its entirety. This application also relates to U.S. application Ser. No. 15/983,632 filed May 18, 2018 and incorporated herein by reference in its entirety. This application also relates to U.S. application Ser. No. 15/983,655 filed May 18, 2018 and incorporated herein by reference in its entirety.

BACKGROUND

[0002] Blockchain usage is growing. As cryptographic blockchain gains acceptance, improved techniques are needed for executing digital contracts.

BRIEF DESCRIPTION OF THE SEVERAL VIEWS OF THE DRAWINGS

[0003] The features, aspects, and advantages of the exemplary embodiments are understood when the following Detailed Description is read with reference to the accompanying drawings, wherein:

[0004] FIGS. **1-8** are simplified illustrations of entity-based cryptographic coinage, according to exemplary embodiments;

[0005] FIGS. **9-11** are detailed illustrations of an operating environment, according to exemplary embodiments;

[0006] FIGS. **12-16** illustrate a blockchain data layer, according to exemplary embodiments;

[0007] FIGS. **17-19** illustrate a digital contract, according to exemplary embodiments;

[0008] FIGS. **20-22** illustrate a filling station, according to exemplary embodiments;

[0009] FIG. **23** illustrates a public entity, according to exemplary embodiments;

[0010] FIGS. **24-25** illustrate virtual computing, according to exemplary embodiments;

[0011] FIG. **26** is a flowchart illustrating a method or algorithm for virtual processing of digital contracts, according to exemplary embodiments; and

[0012] FIGS. **27-28** depict still more operating environments for additional aspects of the exemplary embodiments.

DETAILED DESCRIPTION

[0013] The exemplary embodiments will now be described more fully hereinafter with reference to the accompanying drawings. The exemplary embodiments may, however, be embodied in many different forms and should not be construed as limited to the embodiments set forth herein. These embodiments are provided so that this disclosure will be thorough and complete and will fully convey the exemplary embodiments to those of ordinary skill in the art. Moreover, all statements herein reciting embodiments, as well as specific examples thereof, are intended to encompass both structural and functional equivalents thereof. Additionally, it is intended that such equivalents include both currently known equivalents as well as equivalents developed in the future (i.e., any elements developed that perform the same function, regardless of structure).

[0014] Thus, for example, it will be appreciated by those of ordinary skill in the art that the diagrams, schematics, illustrations, and the like represent conceptual views or processes illustrating the exemplary embodiments. The functions of the various elements shown in the figures may be provided through the use of dedicated hardware as well as hardware capable of executing associated software. Those of ordinary skill in the art further understand that the exemplary hardware, software, processes, methods, and/or operating systems described herein are for illustrative purposes and, thus, are not intended to be limited to any particular named manufacturer.

[0015] As used herein, the singular forms "a," "an," and "the" are intended to include the plural forms as well, unless expressly stated otherwise. It will be further understood that the terms "includes," "comprises," "including," and/or "comprising," when used in this specification, specify the presence of stated features, integers, steps, operations, elements, and/or components, but do not preclude the presence or addition of one or more other features, integers, steps, operations, elements, components, and/or groups thereof. It will be understood that when an element is referred to as being "connected" or "coupled" to another element, it can be directly connected or coupled to the other element or intervening elements may be present. Furthermore, "connected" or "coupled" as used herein may include wirelessly connected or coupled. As used herein, the term "and/or" includes any and all combinations of one or more of the associated listed items.

[0016] It will also be understood that, although the terms first, second, etc. may be used herein to describe various elements, these elements should not be limited by these terms. These terms are only used to distinguish one element from another. For example, a first device could be termed a second device, and, similarly, a second device could be termed a first device without departing from the teachings of the disclosure.

[0017] FIGS. **1-8** are simplified illustrations of entity-based cryptographic coinage **20**, according to exemplary embodiments. Here any entity **22** may create its own cryptographic coinage **20** in a blockchain environment **24**. The entity **22**, in other words, may establish entity-specific electronic tokens **26** to access and/or to use the blockchain environment **24**. While exemplary embodiments may be applied to any entity **22**, most readers are thought familiar with financial services. That is, suppose the entity **22** is a bank, lender, or other financial institution **28** (such as PIMCO®, CITI®, or BANK OF AMERICA®). As the reader likely understands, the financial institution **28** creates a massive amount of banking records, transaction records, mortgage instruments, and other private data **30**. The financial institution **28** thus has a software application **32** that encrypts its private data **30**. While the financial institution **28** may use any encryption scheme, FIG. **1** illustrates a private blockchain **34**. That is, the financial institution **28** cryptographically hashes the private data **30** into the private blockchain **34** and sends or feeds the private blockchain **34** to a blockchain data layer **36**. The blockchain data layer **36**

generates various data records **38**, as later paragraphs will explain. Moreover, the blockchain data layer **36** may also add another layer of cryptographic hashing to generate a public blockchain **40**. The blockchain data layer **36** acts as a validation service **42** and generates a cryptographic proof **44**. The public blockchain **40** thus publishes the cryptographic proof **44** as a public ledger **46** that establishes chains of blocks of immutable evidence.

[0018] The entity-specific tokens **26** are associated with the entity **22**. The financial institution **28**, for example, generates and/or issues the entity-specific tokens **26** to access the private blockchain **34**. Because the private blockchain **34** represents hashes of the financial institution's private data **30**, the private blockchain **34** may be considered a private resource or property of the financial institution **28**. That is, the private blockchain **34** is controlled by, or affiliated with, the financial institution **28**, so the financial institution **28** may control who adds and/or writes to the private blockchain **34** and who reads, accesses, or receives the private blockchain **34**.

[0019] The entity-specific tokens **26** may thus be control mechanisms. While the entity-specific tokens **26** may have any functional scheme, FIG. **1** illustrates a private credit token **50** and a private tradeable token **52**. The entity's credit token **50**, for example, may be acquired and then spent or burned when accessing the financial institution's private blockchain **34**. The entity's credit token **50**, in other words, represents any credit-based entry system associated with the financial institution's private blockchain **34**. The tradeable token **52**, on the other hand, may be generated for transfer among others. The entity **22** generates the tradeable token **52** to be traded and/or spent. The tradeable token **52**, in other words, may be considered as the entity's specific, private currency to be used as the entity **22** governs.

[0020] Moreover, the blockchain **34** may further reference a digital contract **54**. The digital contract **54** is a software program that adds one or more layers of information onto digital transactions being executed by or on the blockchain **34**. The digital contract **54** is sometimes referred to as a self-executing or "smart" contract between parties to a transaction. When the digital contract **54** is executed, the parties to the digital contract **54** may be compensated. While there are many compensation schemes, most readers are perhaps familiar with crypto-compensation. That is, when the digital contract **54** successfully executes, perhaps the parties exchange, trade, or transfer the credit token **50** and/or the tradeable token **52**. When any party, or all the parties, perform their assigned role in the transaction, value is given via the credit token **50** and/or the tradeable token **52**.

[0021] The digital contract **54** is thus a computer program or code that verifies and/or enforces negotiation and/or performance of a contract between parties. One fundamental purpose of so-called smart contracts is to integrate the practice of contract law and related business practices with electronic commerce protocols between parties or devices via the Internet. Smart contracts may leverage a user interface that provides one or more parties or administrators access, which may be restricted at varying levels for different people, to the terms and logic of the contract. Smart contracts typically include logic that emulates contractual clauses that are partially or fully self-executing and/or self-enforcing. Examples of smart contracts are digital rights management (DRM) used for protecting copyrighted works, financial cryptography schemes for financial contracts, admission control schemes, token bucket algorithms, other quality of service mechanisms for assistance in facilitating network service level agreements, person-to-person network mechanisms for ensuring fair contributions of users, and others. Smart contract infrastructure can be implemented by replicated asset registries and contract execution using cryptographic hash chains and *Byzantine* fault tolerant replication. For example, each node in a peer-to-peer network or blockchain distributed network may act as a title registry and escrow, thereby executing changes of ownership and implementing sets of predetermined rules that govern transactions on the network. Each node may also check the work of other nodes and in some cases, as noted above, function as miners or validators.

[0022] FIGS. **2-3** illustrate examples of the entity-specific tokens **26**. Suppose that a third-party **60** wishes to receive, read, write to, or otherwise access the financial institution's private blockchain **34** and/or the digital contract **54**. As FIG. **2** illustrates, exemplary embodiments may require that the third-party **60** spend or burn one or more of the credit tokens **50**. The credit token **50** may thus control access to the financial institution's private blockchain **34** and/or the digital contract **54**. The inventor envisions that vendors, service providers, individual users, and other third-parties **60** may wish to access the hash values of the private data **30** contained within the financial institution's private blockchain **34**. Moreover, the third party may want to access, inspect, execute, or verify the digital contract **54**. The financial institution **28** may thus require that the third-party **60** redeem the entity's credit token(s) **50** before granting read, write, or access permission to the digital contract **54**. The financial institution **28** may additionally or alternatively require redemption of the entity's credit token(s) **50** for using protocols, rules, and application programming interfaces ("APIs") associated with the private blockchain **34** and/or the digital contract **54**. The financial institution **28** may thus establish or issue its own credit tokens **50** and even govern their usage restrictions **62** and value **64**, as later paragraphs will explain.

[0023] FIG. **3** illustrates the tradeable token **52**. The financial institution **28** may establish the tradeable token **52** and also govern its usage restrictions **62** and value **64**. The tradeable token **52**, in other words, is a cryptocurrency or "coin." Again, while exemplary embodiments may utilize any functional scheme, the tradeable token **52** may be earned. That is, anyone (such as the third party **60**) may earn the tradeable token **52** according to the usage restrictions **62**. For example, suppose the blockchain data layer **36** earns the entity's tradeable token(s) **52** in exchange for the validation service **42**. That is, a provider of the validation service **42** is paid, or earns, the entity's tradeable token(s) **52** for processing or executing the digital contract **54** and/or for cryptographically hashing the proof **44** of the digital contract **54**. The provider of the validation service **42** may also be paid in the entity's tradeable token(s) **52** for publishing the proof **44**. The tradeable token **52** may thus be transferred as currency according to the usage restrictions **62** and its value **64**.

[0024] FIG. **4** illustrates transaction records **70**. Whenever the entity-specific tokens **26** are created, owned, or transferred, the transaction record **70** may be generated. The transaction record **70** may then be documented in the blockchain environment **24**. For example, the entity-specific tokens **26** may be addressable. That is, the credit token **50**

and the tradeable token **52** may be uniquely associated with a common, single cryptographic address **72**. The cryptographic address **72** may represent an owner or holder (e.g., the entity **22** or the third-party **60**). When the entity-specific tokens **26** are created, generated, or assigned, the entity-specific tokens **26** may be assigned or associated with the cryptographic address **72**. The cryptographic address **72** may then be received by, and propagated within, the blockchain data layer **36** to identify the corresponding data records **38**. The blockchain data layer **36** may even hash the cryptographic address **72** as the cryptographic proof **44** of the transaction records **70**. Exemplary embodiments thus publically document the transaction records **70** involving the entity-specific tokens **26**, based on the single cryptographic address **72**. In simple words, the blockchain data layer **36** publishes ownership and transfer proofs **44** of the credit token **50** and the tradeable token **52** based on the transaction records **70** associated with the single cryptographic address **72**.

[0025] The transaction records **70** may also document the digital contract **54**. Whenever the digital contract **54** is generated, processed, or even executed, the transaction record **70** may be generated. The transaction record **70** may then be documented in the blockchain environment **24**. For example, the entity-specific tokens **26** may be earned as payment according to the executable terms of the digital contract **54**. The entity-specific tokens **26** may additionally or alternatively be earned or awarded for processing or executing a portion of, or entirely, the digital contract **54**. The entity-specific tokens **26** may thus be uniquely associated with a party to the digital contract **54** and/or with a service provider/processor of the digital contract **54**. The transaction record **70** may document the parties to the digital contract **54**, a transactional description describing a transaction governed by the digital contract **54**, and any financial or performance terms. The transaction record **70** may thus document an offer, an acceptance, a consideration, and terms. For simplicity, then, the single cryptographic address **72** may represent a party to the digital contract **54** and/or with a service provider/processor of the digital contract **54**. Regardless, when the entity-specific tokens **26** are created, generated, or assigned, the entity-specific tokens **26** may be received by, and propagated within, the blockchain data layer **36** to identify the corresponding data records **38**. The blockchain data layer **36** may thus publish the proofs **44** of the digital contract **54** and any entity-specific tokens **26** paid or exchanged, according to the transaction records **70**.

[0026] FIG. **5** illustrates a filling station **80** in the blockchain environment **24**. Because the tokens **26** may be consumed by users (such as during or after any processing or execution of the digital contract **54**), the filling station **80** allows the third party **60** to replenish or fill an account **82**. Recall that the third-party entity **22** may be required to spend the tokens **26** to access the financial institution's private blockchain **34** and/or the digital contract **54**. Moreover, the tokens **26** may also be earned or transferred according to the terms of the digital contract **54**. The account **82** may thus be established, and the account **82** maintains a monetary or numerical balance **84** of the tokens **26**. As the tokens **26** are spent, traded, or redeemed, the account **82** may need filling to continue using or accessing the blockchain **34** and/or the digital contract **54**.

[0027] The filling station **80** may access both the transaction records **70** and the blockchain data layer **36**. Because the blockchain data layer **36** may document the data records **38** using the single cryptographic address **72**, the single cryptographic address **72** may serve as a common reference or query parameter with the entity's transaction records **70**. The filling station **80**, in other words, may use the single cryptographic address **72** to identify the transaction records **70** that correspond to the blockchain data layer **36**. The filling station **80** may thus present a transaction summary of the account **82** and the balance **84**. Because blockchain data layer **36** may track and/or prove the transaction records **70**, exemplary embodiments may search the blockchain data layer **36** for the single cryptographic address **72**. That is, the filling station **80** may query the blockchain data layer **36** for the single cryptographic address **72**, and the blockchain data layer **36** may identify the transaction records **70** that match the single cryptographic address **72**. The filling station **80** may then process the transaction records **70** to provide the transaction summary of the account **82**, the balance **84**, and any other transactional data. The filling station **80** may also allow the user to replenish an amount or value of the tokens **26**, thus allowing the user to continue exchanging the tokens **26** for access to the private blockchain **34**, the blockchain data layer **36**, and/or the digital contract **54**.

[0028] FIG. **6** further illustrates the filling station **80**. Here the blockchain data layer **36** may have its own cryptocoinage **90**. That is, a provider of the blockchain data layer **36** may establish its cryptocoinage **90** for accessing and/or using the validation service **42**. The cryptocoinage **90** may thus include a credit token and a tradeable token (not shown for simplicity). The credit token may be required to enter or access the blockchain data layer **36** to receive the validation service **42**, and the tradeable token may be earned for participating in the validation service **42**. Regardless, the filling station **80** may use the single cryptographic address **72**. The third party **60** may use the single cryptographic address **72** to access the entity's cryptocoinage **20** and the blockchain data layer's cryptocoinage **90**. Exemplary embodiments may thus identify and track the transaction records **70** and the blockchain data layer's cryptocoinage **90** using the same, single cryptographic address **72**.

[0029] Exemplary embodiments thus present an elegant solution. Any entity **22** may create its own private blockchain **34** and offer or present the digital contract **54** for self-execution. The entity **22** may then establish or create the tokens **26** for using, accessing, or processing the entity's private blockchain **34** and/or the digital contract **54**. The tokens **26** may have the value **64**, thus fostering a market for entity-specific tradeable assets in the blockchain environment **24**. The tradable value **64** of the tokens **26** may thus drive demand to use the digital contracts **54**. Exemplary embodiments may thus provide a two-token system that isolates any use of the entity's private blockchain **34** from the entity's tradeable token **52**. Moreover, the credit token **50** may be associated with the third party **60** (perhaps via the single cryptographic address **72**), thus allowing the third party **60** to retrieve the account balance **84** from the filling station **80** and sign entries or other transactions. Moreover, the third party **60** may also use the single cryptographic address **72** to access the blockchain data layer **36** via the filling station **80**. The filling station **80** is a single resource or destination (such as a secure web site) for managing a user's cryptographic coinage **20** and defining payments according to the digital contract **54**.

[0030] FIG. **7** expands the entity concept. Here multiple, different entities **22***a-d* provide their respective software applications **32***a-d* that encrypt their respective private data **30***a-d* as their individual, private blockchains **34***a-d*. While exemplary embodiments may be applied to any number of industries or services, FIG. **7** illustrates a simple example of four (4) different entities **22***a-d*. First entity **22***a*, for example, again represents the bank, lender, or other financial institution **28** that encrypts its private data **30***a* as its private blockchain **34***a*. Second entity **22***b* represents any retailer **90** (such as HOME DEPOT®, KOHL'S®, or WALMART®) that encrypts its private data **30***b* as its private blockchain **34***b*. Third entity **22***c* represents a website **92** offering a service **94** (such as AMAZON®, NETFLIX®, or GOOGLE®) that encrypts its private data **30***c* as the private blockchain **34***c*. Fourth entity **22***d* represents an automotive or other manufacturer or supplier **96** (such as FORD®, TOYOTA®, or DELPHI®) that encrypts its private data **30***d* as the private blockchain **34***d*. The entities **22***a-d* thus use their respective software applications **32***a-d* to provide a first layer **98** of cryptographic hashing. The entities **22***a-d* may also use their respective software applications **32***a-d* to issue their own private and entity-specific cryptocoinage **20***a-d*. Each entity **22***a-d* may then send their respective private blockchains **34***a-d* to the blockchain data layer **36**, and the blockchain data layer **36** may add a second layer **100** of cryptographic hashing. The blockchain data layer **36** thus generates the public blockchain **40** as a public resource or utility for record keeping. Any entity **22** that subscribes to the blockchain data layer **36** (such as by acquiring and/or spending the cryptocoinage **90**. Any entity **22** may thus write and store the proofs **44** of its private data **30** to the public blockchain **40**. The blockchain data layer **36**, in other words, acts as the public ledger **46** that establishes chain of blocks of immutable evidence.

[0031] As FIG. **7** also illustrates, each entity **22***a-d* may establish its own private cryptocoinage **20***a-d*. Each entity's private software application **32***a-d* may create and/or issue its cryptocoinage **20***a-d* (such as respective entity-specific tokens **26** above explained). Each entity **22***a-d* may also establish its own usage restrictions and value (illustrated as reference numerals **62** and **64** in FIGS. **2-3**) according to rules governing ownership, trade, and other policies. Each entity **22***a-d* may generate and sends its respective transaction records **70***a-d* which reference each entity's single cryptographic address **72***a-d*) to the blockchain data layer **36** for documentation.

[0032] As FIG. **7** further illustrates, each entity **22***a-d* may also specify their respective digital contract **54***a-d*. When any of the private blockchains **34***a-d* is received, the blockchain data layer **36** may coordinate execution of any digital contract **54***a-d*. The blockchain data layer **36**, for example, may inspect any private blockchain **34***a-d* and identify any information associated with the digital contract **54***a-d*. The blockchain data layer **36** may then execute the digital contract **54***a-d*, and/or the blockchain data layer **36** may identify a service provider that executes the digital contract **54***a-d*. The blockchain data layer **36**, in other words, may manage the execution of the digital contracts **54***a-d* according to a subcontractor relationship. A provider of the blockchain data layer **36** may then be compensated via any entity's cryptocoinage **20***a-d* and/or the blockchain data layer's cryptocoinage **90**.

[0033] As FIG. **8** illustrates, the filling station **80** is agnostic. Any user (such as the entity **22***a-d* or the third party **60**) may authenticate to the filling station **80**. Once authenticated, the user need only enter or provide the correct single cryptographic address **72***a-d* to access the entity's private cryptocoinage **20***a-d*, the blockchain data layer's cryptocoinage **90**, and/or the entity's digital contract **54***a-d*. The single cryptographic address **72***a-d*, in other words, allows the user to access her account **82** and balance **84** for the entity's private cryptocoinage **20***a-d*, the blockchain data layer's cryptocoinage **90**, and/or the entity's digital contract **54***a-d*. The user may thus easily conduct transactions between the entity's private cryptocoinage **20***a-d* and the blockchain data layer's cryptocoinage **90**. The entity **22***a-d*, for example, may fuel or replenish its supply of the blockchain data layer's cryptocoinage **90**, perhaps by redeeming or exchanging the entity's private cryptocoinage **20***a-d* (perhaps according to an exchange rate or other value). Similarly, the provider of the blockchain data layer **36** may fuel or replenish its supply of the entity's private cryptocoinage **20***a-d* by purchasing or exchanging the blockchain data layer's cryptocoinage **90**. The provider of the blockchain data layer **36** may also earn the entity's private cryptocoinage **20***a-d* by processing any portion of, or by executing, the entity's digital contract **54***a-d*. Moreover, the respective private blockchains **34***a-d* and the blockchain data layer **36** would contain the data records **38** confirming the processing and/or execution of the digital contract **54***a-d*, so the transaction records **70***a-d* thus propagate into the blockchain data layer **36** for public disclosure via the public blockchain **40**. Any user that successfully authenticates to the filling station **80** may access a full accounting of his or her digital cryptocoinages **20***a-d* and/or **90** and any digital contracts **54**, perhaps according to the respective single cryptographic address **72***a-d*. The user may thus buy, sell, trade, and/or redeem any entity-specific cryptocoinages **20***a-d* and/or **90**, all by accessing the filling station **80**. The user may buy or sell any entity's coins or replenish credits, all by accessing the filling station **80**. The user may also track performance or obligations defined by the digital contracts **54***a-d* and any payments or consideration received or paid.

[0034] Exemplary embodiments thus present another elegant solution. The filling station **80** is another service offered by the blockchain data layer **36**. Because all the transaction records **70** in the blockchain data layer **36** are identifiable (perhaps via the single cryptographic address **72**), the filling station **80** can present the summary of the user's credit tokens and tradeable tokens. The filling station **80** may thus provide a single or universal electronic wallet for all of a user's digital coinage and credits, regardless of the issuing entity **22***a-d*. The user may thus only perform a single authentication to the blockchain data layer **36** and access all her cryptofunds.

[0035] FIGS. **9-11** are more detailed illustrations of an operating environment, according to exemplary embodiments. FIG. **8** illustrates an entity server **110** communicating with a data layer server **112** via a communications network **114**. The entity server **110** operates on behalf of the entity **22** and generates the entity's private blockchain **34**. The entity server **110**, in other words, has a processor **116** (e.g., "μP"), application specific integrated circuit (ASIC), or other component that executes the entity's software application **32** stored in a local memory device **118**. The entity server **110**

has a network interface to the communications network **114**, thus allowing two-way, bidirectional communication with the data layer server **112**. The entity's software application **32** includes instructions, code, and/or programs that cause the entity server **110** to perform operations, such as calling, invoking, and/or applying an electronic representation of a hashing algorithm **120** to the entity's private data **30**. The hashing algorithm **120** thus generates one or more hash values **122**, which are incorporated into the entity's private blockchain **34**. The entity's software application **32** then instructs the entity server **110** to send the private blockchain **34** via the communications network **114** to a network address (e.g., Internet protocol address) associated with the data layer server **112**.

[0036] The digital contract **54** may also be identified. The entity's software application **32** may also instruct the entity server **110** to include the digital contract **54** as informational content in the private blockchain **34**. For example, the digital contract **54** may be identified by a contract identifier **124** and contractual information **126**. The contract identifier **124** is any digital identifying information that uniquely identifies or references the digital contract **54**. The contract identifier **124** may be an alphanumeric combination that uniquely identifies a vendor and/or version of the digital contract **54** and/or a processor or executioner of the digital contract **54**. The contract identifier **124** may also be one of the unique hash values **122** (perhaps generated by the hashing algorithm **120**) that is included within, or specified by, the private blockchain **34**. Similarly, the contractual information **126** may identify the parties to the digital contract **54**, their respective performance obligations and terms, and consideration.

[0037] FIG. **10** illustrates the blockchain data layer **36**. The data layer server **112** has a processor **130** (e.g., "μP"), application specific integrated circuit (ASIC), or other component that executes a data layer application **132** stored in a local memory device **134**. The data layer server **112** has a network interface to the communications network **114**. The data layer application **132** includes instructions, code, and/or programs that cause the data layer server **112** to perform operations, such as receiving the entity's private blockchain **34**, the digital contract **54**, the contract identifier **124**, and/or the contractual information **126**. The data layer application **132** then causes the data layer server **112** to generate the blockchain data layer **36**. The data layer application **132** may optionally call, invoke, and/or apply the hashing algorithm **120** to the data records **38** contained within the blockchain data layer **36**. The data layer application **132** may also generate the public blockchain **40**. The data layer application **132** may thus generate the public ledger **46** that publishes, records, or documents the digital contract **54**, the contract identifier **124**, and/or the contractual information **126**. Indeed, if the data layer application **132** processes and/or manages the digital contract **54**, the data records **38** may document any processing or execution, and the data layer application **132** may optionally apply the hashing algorithm **120** to the data records **38** to generate the cryptographic proof **44** of the digital contract **54**.

[0038] FIG. **11** illustrates additional publication mechanisms. Once the blockchain data layer **36** is generated, the blockchain data layer **36** may be published in a decentralized manner to any destination. The data layer server **112**, for example, may generate and distribute the public blockchain **40** (via the communications network **114** illustrated in FIGS. **9-10**) to one or more federated servers **140**. While there may be many federated servers **140**, for simplicity FIG. **11** only illustrates two (2) federated servers **140***a* and **140***b*. The federated servers **140***a* and **140***b* provide a service and, in return, they are compensated according to a compensation or services agreement or scheme.

[0039] Exemplary embodiments include still more publication mechanisms. For example, the cryptographic proof **44** and/or the public blockchain **40** may be sent (via the communications network **114** illustrated in FIGS. **9-10**) to a server **142**. The server **142** may then add a another, third layer of cryptographic hashing (perhaps using the hashing algorithm **120**) and generate another or second public blockchain **144**. While the server **142** and/or the public blockchain **144** may be operated by, or generated for, any entity, exemplary embodiments may integrate another cryptographic coin mechanism. That is, the server **142** and/or the public blockchain **144** may be associated with BITCOIN®, ETHEREUM®, RIPPLE®, or other cryptographic coin mechanism. The cryptographic proof **44** and/or the public blockchain **40** may be publically distributed and/or documented as evidentiary validation. The cryptographic proof **44** and/or the public blockchain **40** may thus be historically and publically anchored for public inspection and review.

[0040] Exemplary embodiments may be applied regardless of networking environment. Exemplary embodiments may be easily adapted to stationary or mobile devices having cellular, wireless fidelity (WI-FI®), near field, and/or BLUETOOTH capability. Exemplary embodiments may be applied to mobile devices utilizing any portion of the electromagnetic spectrum and any signaling standard (such as the IEEE 802 family of standards, GSM/CDMA/TDMA or any cellular standard, and/or the ISM band). Exemplary embodiments, however, may be applied to any processor-controlled device operating in the radio-frequency domain and/or the Internet Protocol (IP) domain. Exemplary embodiments may be applied to any processor-controlled device utilizing a distributed computing network, such as the Internet (sometimes alternatively known as the "World Wide Web"), an intranet, a local-area network (LAN), and/or a wide-area network (WAN). Exemplary embodiments may be applied to any processor-controlled device utilizing power line technologies, in which signals are communicated via electrical wiring. Indeed, exemplary embodiments may be applied regardless of physical componentry, physical configuration, or communications standard(s).

[0041] Exemplary embodiments may utilize any processing component, configuration, or system. Any processor could be multiple processors, which could include distributed processors or parallel processors in a single machine or multiple machines. The processor can be used in supporting a virtual processing environment. The processor could include a state machine, application specific integrated circuit (ASIC), programmable gate array (PGA) including a Field PGA, or state machine. When any of the processors execute instructions to perform "operations," this could include the processor performing the operations directly and/or facilitating, directing, or cooperating with another device or component to perform the operations.

[0042] Exemplary embodiments may packetize. When the entity server **110** and the data layer server **112** communicate via the communications network **114**, the entity server **110** and the data layer server **112** may collect, send, and retrieve information. The information may be formatted or generated

as packets of data according to a packet protocol (such as the Internet Protocol). The packets of data contain bits or bytes of data describing the contents, or payload, of a message. A header of each packet of data may contain routing information identifying an origination address and/or a destination address.

[0043] FIGS. **12-16** further illustrate the blockchain data layer **36**, according to exemplary embodiments. The blockchain data layer **36** chains hashed directory blocks **150** of data into the public blockchain **40**. For example, the blockchain data layer **36** accepts input data (such as the entity's private blockchain **34** illustrated in FIGS. **1-10**) within a window of time. While the window of time may be configurable from fractions of seconds to hours, exemplary embodiments use ten (10) minute intervals. FIG. **12** illustrates a simple example of only three (3) directory blocks **150***a-c* of data, but in practice there may be millions or billions of different blocks. Each directory block **150** of data is linked to the preceding blocks in front and the following or trailing blocks behind. The links are created by hashing all the data within a single directory block **150** and then publishing that hash value within the next directory block.

[0044] As FIG. **13** illustrates, published data may be organized within chains **152**. Each chain **152** is created with an entry that associates a corresponding chain identifier **154**. Each entity **22***a-f*, in other words, may have its corresponding chain identifier **154***a-d*. The blockchain data layer **36** may thus track any data associated with the entity **22***a-f* with its corresponding chain identifier **154***a-d*. New and old data in time may be associated with, linked to, identified by, and/or retrieved using the chain identifier **154***a-d*. Each chain identifier **154***a-d* thus functionally resembles a directory **156***a-d* (e.g., files and folders) for organized data entries according to the entity **22***a-f*.

[0045] FIG. **14** illustrates the data records **38** in the blockchain data layer **36**. As data is received as an input (such as the private blockchain **34** and/or the digital contract **54** illustrated in FIGS. **1-10**), data is recorded within the blockchain data layer **36** as an entry **160**. While the data may have any size, small chunks (such as 10 KB) may be pieced together to create larger file sizes. One or more of the entries **160** may be arranged into entry blocks **162** representing each chain **152** according to the corresponding chain identifier **154**. New entries for each chain **152** are added to their respective entry block **162** (again perhaps according to the corresponding chain identifier **154**). After the entries **160** have been made within the proper entry blocks **162**, all the entry blocks **162** are then placed within in the directory block **150** generated within or occurring within a window **164** of time. While the window **164** of time may be chosen within any range from seconds to hours, exemplary embodiments may use ten (10) minute intervals. That is, all the entry blocks **162** generated every ten minutes are placed within in the directory block **150**.

[0046] FIG. **15** illustrates cryptographic hashing. The data layer server **112** executes the data layer application **132** to generate the data records **38** in the blockchain data layer **36**. The data layer application **132** may then instruct the data layer server **112** to execute the hashing algorithm **120** on the data records **38** (such as the directory block **150** illustrated in FIGS. **12** & **14**). The hashing algorithm **120** thus generates one or more hash values **166** as a result, and the hash values **166** represent the hashed data records **38**. As one example, the blockchain data layer **36** may apply a Merkle tree analysis to generate a Merkle root (representing a Merkle proof **44**) representing each directory block **150**. The blockchain data layer **36** may then publish the Merkle proof **44** (as this disclosure explains).

[0047] FIG. **16** illustrates hierarchical hashing. The entity's private software application **32** provides the first layer **98** of cryptographic hashing and generates the private blockchain **34**. The entity **22** then sends its private blockchain **34** (perhaps referencing or specifying the digital contract **54**) to the data layer server **112**. The data layer server **112**, executing the data layer application **132**, generates the blockchain data layer **36**. The data layer application **132** may optionally provide the second or intermediate layer **100** of cryptographic hashing to generate the cryptographic proof **44**. The data layer application **132** may also publish any of the data records **38** as the public blockchain **40**, and the cryptographic proof **44** may or may not also be published via the public blockchain **40**. The public blockchain **40** and/or the cryptographic proof **44** may be optionally sent to the server **142** as an input to yet another public blockchain **144** (again, such as BITCOIN®, ETHEREUM®, or RIPPLE®) for a third layer **170** of cryptographic hashing and public publication. The first layer **98** and the second layer **100** thus ride or sit atop a conventional public blockchain **144** (again, such as BITCOIN®, ETHEREUM®, or RIPPLE®) and provide additional public and/or private cryptographic proofs **44**.

[0048] Exemplary embodiments may use any hashing function. Many readers may be familiar with the SHA-256 hashing algorithm. The SHA-256 hashing algorithm acts on any electronic data or information to generate a 256-bit hash value **64** as a cryptographic key. The key is thus a unique digital signature. There are many hashing algorithms, though, and exemplary embodiments may be adapted to any hashing algorithm.

[0049] FIGS. **17-19** are more detailed illustrations of the digital contract **54**, according to exemplary embodiments. The private entity **22** sends its private blockchain **34** to the network address associated with the data layer server **112** that generates the blockchain data layer **36**. The private blockchain **34** may contain information representing the transaction records **70** associated with the entity's private cryptocoinage **20** (perhaps as one or more privately hashed blocks **180** of data). The private blockchain **34** may also specify, or incorporate, information or data representing the single cryptographic address **72** and/or the digital contract **54** (e.g., the contract identifier **124** and the contractual information **126**, as explained with reference to FIGS. **9-10**). The single cryptographic address **72** and/or the digital contract **54** (e.g., the contract identifier **124** and the contractual information **126**) may additionally or alternatively be separately sent from the entity server **110** to the data layer server **112**. Regardless, the entity's private cryptocoinage **20** may be associated with the digital contract **54** and/or the single cryptographic address **72**. The transaction records **70** and/or the privately hashed blocks **180** of data may thus specify, include, reference, and/or be associated with, and/or identified by, the single cryptographic address **72**, the digital contract **54**, the contract identifier **124**, and/or the contractual information **126**). Because the contract identifier **124** (and/or its corresponding hash value) is an identifiable input to the data layer server **112** generating the blockchain data layer **36**, the data records **38** may also carry or reference the contract identifier **124**. So, should the blockchain data layer **36** create or issue its own cryptocoinage **90**, the cryptocoin-

age **90** may also reference, be identified by, or be associated with the single cryptographic address **72** and/or the contract identifier **124**. The single cryptographic address **72** and/or the contract identifier **124** may thus common indicators or reference data for tracking both the entity's private cryptocoinage **20** and the cryptocoinage **90** issued by the blockchain data layer **36**, according to the terms of the digital contract **54**. The transaction records **70** (representing entity's private cryptocoinage **20**) may thus be commonly mapped or identified to the cryptocoinage **90** issued by the blockchain data layer **36** and to the digital contract **54**.

[0050] FIG. **18** illustrates a simple illustration. Once the contract identifier **124** (and/or its corresponding hash value) is received, the contract identifier **124** may propagate and be recorded within the blockchain data layer **36**. The contract identifier **124**, for example, may be recorded in any of the entries **160**. The entry **160**, and thus the contract identifier **124**, may then be recorded and/or arranged as the entry block **162** and placed within the directory block **150**. The entry **160**, the entry block **162**, and the directory block **150** may thus reference, specify, or be associated with, the contract identifier **124**. The contract identifier **124** has thus propagated as informational content from the private blockchain **34** and into and through the blockchain data layer **36**. The contract identifier **124** thus hierarchically moves through the multiple layers of cryptographic hashing for public publication. The blockchain data layer **36** thus tracks the transaction records **70** involving the contract identifier **124**. In simple words, the blockchain data layer **36** may track contractual performance of the digital contract **54** via the transaction records **70** that reference or contain the contract identifier **124**. Moreover, the blockchain data layer **36** may also track ownership and transfer of the entity's private cryptocoinage **20** and the cryptocoinage **90** issued by the blockchain data layer **36**, all via the common single cryptographic address **72** and/or the contract identifier **124**.

[0051] FIG. **19** illustrates more details. While the single cryptographic address **72** and/or the contract identifier **124** may be any alphanumeric entry or biometric input, FIG. **19** illustrates a common authentication mechanism **190**. Here the same or similar authentication mechanism **190** is used to access both the entity's private cryptocoinage **20** and the cryptocoinage **90** issued by the blockchain data layer **36**. If a user of the blockchain data layer **36** satisfies the authentication mechanism **190**, then exemplary embodiments may access both the private cryptocoinage **20**, the cryptocoinage **90**, and/or the data records **38** associated with the contract identifier **124**. As a simple example, suppose the user of the authentication mechanism **190** supplies information or data representing the single cryptographic address **72** and/or the contract identifier **124**. The single cryptographic address **72** and/or the contract identifier **124** may be any unique alphanumeric entry, biometric input, user identifier, or other authentication credential. For example, most readers are likely familiar with an alphanumeric username and password, which is a common authentication mechanism **190**. FIG. **19**, though, illustrates a passphrase **192** (such as a multi-word mnemonic). When the entity's private cryptocoinage **20** is/are created, generated, or assigned, the entity's private cryptocoinage **20** may be assigned or associated with the passphrase **192**. The passphrase **192** is unique to the registered owner, possessor, or user of the entity's private cryptocoinage **20**. The passphrase **192** may even be hashed as a hash value and supplied to the blockchain data layer **36** (as above explained). The passphrase **192**, in other words, may be hashed as the single cryptographic address **72** and propagated within the blockchain environment **24** to document the transaction records **70** involving the entity's private cryptocoinage **20**.

[0052] The passphrase **192** may also authenticate to the cryptocoinage **90**. If the user correctly supplies the passphrase **192**, then the same user may conduct transactions involving the cryptocoinage **90** issued by the blockchain data layer **36** and/or involving the contract identifier **124** associated with the digital contract **54**. Exemplary embodiments thus allow the user to order transactions and exchanges involving the entity's private cryptocoinage **20**, the cryptocoinage **90** issued by the blockchain data layer **36**, and/or the digital contract **54**.

[0053] FIGS. **20-22** further illustrate the filling station **80**, according to exemplary embodiments. The filling station **80** may be a public and/or private service for financial transactions involving the entity's private cryptocoinage **20**, the cryptocoinage **90** issued by the blockchain data layer **36**, and/or the digital contract **54**. FIG. **20** illustrates the filling station **80** as a software-as-a-service offered by the secure data layer server **112** for accessing the blockchain data layer **36**. The filling station **80**, for example, may be a module within, or called by, the data layer application **132**. A user accesses the filling station **80** to conduct transactions involving her private cryptocoinage **20**, the cryptocoinage **90** (issued by the blockchain data layer **36**), and/or the digital contract **54**. While the filling station **80** may have any user interface, FIG. **20** illustrates a web interface **194**. That is, the filling station **80** may be accessed via a webpage **196**. The webpage **196** prompts the user to input her authentication credentials according to the authentication mechanism **190** (such as typing the passphrase **192** into a data field or audibly speaking the passphrase **192**).

[0054] FIG. **21** further illustrates the web interface **194**. The user accesses the filling station **80** using a user device **200**. While the user device **200** may be any processor-controlled device, most readers are familiar with a smartphone **202**. If the smartphone **202** correctly sends authentication credentials (such as the single cryptographic address **72** and/or passphrase **192**, as above explained), then the smartphone **202** may utilize the web interface **194** to the data layer server **112** and/or the blockchain data layer **36**. The smartphone **202** executes a web browser and/or a mobile application to send a request **204** specifying an address or domain name associated with or representing the filling station **80**. The web interface **194** to the data layer server **112** thus sends the webpage **196** as a response, and the user's smartphone **202** downloads the webpage **196**. The smartphone **202** has a processor and memory device (not shown for simplicity) that causes a display of the webpage **196** as a graphical user interface (or "GUI") **206** on its display device **208**. The GUI **206** may generate one or more prompts or fields for specifying the authentication mechanism **190** and transactional options. For example, the user preferably enters, speaks, or otherwise provides the passphrase **192**. Exemplary embodiments may or may not hash the authentication passphrase (using the hashing algorithm **120** above explained) to produce or generate a hashed passphrase. Exemplary embodiments may then search the blockchain data layer **36** for the data records **38**. That is, exemplary embodiments may query the blockchain data layer **36** for a query parameter (such as the contract identifier **124** and/or

its hashed value) and the blockchain data layer **36** identifies the data records **38** that match or reference the query parameter. The filling station **80** may then process the data records **38** to provide a transactional summary **210** of the digital contract **54**. The filling station **80** may also allow the user to replenish an amount or value of the private cryptocoinage **20** and/or the cryptocoinage **90**, even allowing the user to continue exchanging the cryptocoinage **20** for access to the blockchain data layer **36**.

[0055] Exemplary embodiments may thus share the common authentication mechanism **190**. If the entity's private software application **32** requires the same passphrase **192** to establish any terms of the digital contract **54**, then the passphrase **192** may have been hashed and recorded within the blockchain data layer **36**. The single cryptographic address **72**, the contract identifier **124**, and/or the passphrase **192** may be associated with the data records **38** representing the digital contract **54**, the private cryptocoinage **20** (issued by the entity **22**), and the cryptocoinage **90** (issued by the blockchain data layer **36**). The filling station **80** may thus identify any of the data records **38** that are commonly associated with the contract identifier **124**, the private cryptocoinage **20** (issued by the entity **22**), and/or the cryptocoinage **90**. The filling station **80** thus allows the user to exchange cryptocoinage **20** and **90** for access to the private blockchain **34** and/or the blockchain data layer **36**.

[0056] FIG. **22** illustrates a query mechanism. Here the data layer server **112** may access a database **220** of data layer records. The database **220** of data layer records provides a referential record of the informational content contained within the blockchain data layer **36**. FIG. **22** illustrates the data layer server **112** locally storing the database **220** of data layer records in its local memory device **134**, but the database **220** of data layer records may be remotely stored and accessed via the communications network **114**. Regardless, the data layer server **112** may query the database **220** of data layer records for the single cryptographic address **72** and/or the contract identifier **124** and identify and/or retrieve any corresponding data records **38**. While the database **220** of data layer records may have any logical structure, FIG. **22** illustrates the database **220** of data layer records as a table **222** that maps, converts, or translates the single cryptographic address **72** and/or the contract identifier **124** to its corresponding entry **160**, entry block **162**, and/or directory block **150** within the blockchain data layer **36**. Whenever the data layer server **112** generates the entry **160**, entry block **162**, and/or directory block **150**, the data layer server **112** may add an entry to the database **220** of data layer records. Over time, then, the database **220** of data layer tracks a comprehensive historical repository of information that is electronically associated with its corresponding contract identifier **124**. The data layer server **112** may then read or retrieve the entry **160**, entry block **162**, and/or directory block **150** containing or corresponding to the contract identifier **124**.

[0057] Exemplary embodiments thus present the entity-specific cryptocoinage **20**. Any entity **22** may create its own private blockchain **34**, establish its entity-specific tokens **26**, and define or offer digital contracts **54**. The entity-specific tokens **26** may or may not have the value **64**. The tradeable token **52**, for example, may have a market value based on supply and/or demand, thus allowing or causing the value **64** of the tradeable token **52** to rise/fall or to increase/decrease, based on market forces. The credit token **50**, however, may have a constant price or value, perhaps set by the entity **22**. The entity-specific tokens **26** may be associated with the contract identifier **124**, thus allowing a faster and simpler accounting scheme for machine executable contractual terms.

[0058] Exemplary embodiments thus create coinage on top of coinage. The hierarchical scheme (explained with reference to FIG. **16**) allows the private entity **22** to establish its private cryptocoinage **20** hierarchically above the traditional BITCOIN®, ETHEREUM®, or RIPPLE® coinage. The entity's private data **30** remains private, but the transaction records **70** may be publicly documented or proved via the traditional BITCOIN®, ETHEREUM®, or RIPPLE® environment. The private entity **22**, in other words, need to worry about or concern itself with public publication. The private entity **22** need only subscribe (e.g., pay for write access) to the blockchain data layer **36**. The digital contract **54** may also be offered, executed, and documented by the transaction records **70**.

[0059] FIG. **23** illustrates a public entity **230**, according to exemplary embodiments. Here exemplary embodiments may be applied to public data **232** generated by the public entity **230**. The public entity **230** may be a city, state, or federal governmental agency, but the public entity **230** may also be a contractor, non-governmental organization, or other actor that acts on behalf of the governmental agency. The public entity **230** operates a public server **234** and applies its software application **236** to its public data **232** to generate its governmental blockchain **238**. The public entity **230** may further generate/issue its cryptocoinage **240** and offer digital contracts **54** for governmental, public services. The data layer server **112** receives the governmental blockchain **238** and generates the blockchain data layer **36**. The data layer server **112** may then generate the public blockchain **40** representing any data records **38** representing the public data **232** and/or the cryptocoinage **240**.

[0060] FIGS. **24-25** illustrate virtual computing, according to exemplary embodiments. Here exemplary embodiments may manage virtual machines (or "VM") **250** that execute the digital contracts **54***a-d*. As the reader may understand, the data layer server **112** may provide virtual computing and/or virtual hardware resources to client devices (such as the entities **22***a-d* sending their blockchains **34***a-d*). The data layer server **112** may lend or share its hardware, computing, and programming resources in a virtual computing environment. The data layer server **112** may thus operate or functions as a virtual, remote resource for executing the digital contracts **54***a-d* and for generating the blockchain data layer **40**. The data layer server **112** may present or operate as one or more of the virtual machines **250**. Each one of the virtual machines **250** may provide its processing or application resource to any digital contract **54**. While there may be any number of the virtual machines **250**, FIG. **24** only illustrates a simple example of four (4) virtual machines **250***a-d*. In other words, the number or instantiations may be several or even many, depending on complexity and resources. Virtual computing is generally known, so this disclosure need not dwell on known details.

[0061] Here, though, contractual execution may be preassigned. Recall that the data layer server **112** may receive many different blockchains **34***a-d*, and each blockchain **34***a-d* may specify or reference a different digital contract **54***a-d*. As the data layer server **112** may provide resources to many different entities **22***a-d* and their respective block-

chains **34***a-d*, optimal management techniques may be desired. That is, as the entities **22***a-d* send their digital contracts **54***a-d* and/or make requests for contract processing, some of the shared resources in the data layer server **112** may be dedicated to particular ones of the digital contracts **54**. The data layer server **112** may assign or distribute the various digital contracts **54***a-d* to a particular virtual machine **250** for processing. As a very simple example, suppose the data layer server **112** has four (4) virtual machines **250***a-d*. Each one of the virtual machines **250***a-d* is dedicated to processing a particular one of the digital contracts **54***a-d*. That is, each digital contract **54***a-d* may be preassigned to a corresponding one of the virtual machines **250***a-d*. As the private blockchains **34***a-d* are received as inputs, the data layer server **112** may inspect the private blockchains **34***a-d* for the contract identifier **124** and determine which corresponding one of the virtual machines **34***a-d* processes or executes the digital contract **54***a-d*. Each digital contract **54***a-d*, in other words, may be assigned to a virtual machine **250***a-d* for dedicated processing.

[0062] FIG. **25** illustrates contractual assignments. Here the data layer server **112** may access an electronic database **252** of virtual machines that defines assignments between the digital contracts **54** and their corresponding virtual machine **250**. While the database **252** may have any logical structure, FIG. **25** illustrates the database **252** as a table **254** that maps, converts, or translates the contract identifier **124** to its corresponding virtual machine **250**. The database **252** may thus be preconfigured or preloaded with entries that assign or associate each virtual machine **250** to its corresponding contract identifier **124**. As the data layer server **112** receives any blockchain **34**, the data layer server **112** may inspect the blockchain **34** for the contract identifier **124**. The data layer server **112** may then query the database **252** for the contract identifier **124** to identify the corresponding virtual machine **250**. The data layer server **112** may thus identify and/or retrieve an address, processor core, identifier, or other indicator assigned to the corresponding virtual machine **250**. The database **252** may optionally contain entries that relate hashed values of the contract identifier **124**. Regardless, once the virtual machine **250** is identified, the data layer server **112** may direct, assign, or outsource the relevant data or information to the virtual machine **250** for processing.

[0063] FIG. **26** is a flowchart illustrating a method or algorithm for virtual processing of the digital contracts **54**, according to exemplary embodiments. The private blockchain **34** is received by the data layer server **112** (Block **300**). The private blockchain **34** is inspected for the contract identifier **124** and/or its hash value (Block **302**). The database **252** is consulted to identify the virtual machine **250** (Block **304**). The virtual machine **250** is instructed to process or execute the digital contract **54** (Block **306**). The data records **38** in the blockchain data layer **36** are generated (Block **308**), and the data records **38** describe the execution of the digital contract **54** by the virtual machine **250**. The data records **38** may be hashed (Block **310**) and incorporated into the public blockchain **34** (Block **312**). When a user successfully authenticates to the filling station **80** (Block **314**), the filling station **80** may access the data records **38** in the blockchain data layer **36** (Block **316**) representing the digital contract **54**.

[0064] FIG. **27** is a schematic illustrating still more exemplary embodiments. FIG. **27** is a more detailed diagram illustrating a processor-controlled device **350**. As earlier paragraphs explained, the entity's private software application **32** and/or the data layer application **132** may partially or entirely operate in any mobile or stationary processor-controlled device. FIG. **27**, then, illustrates the entity's private software application **32** and/or the data layer application **132** stored in a memory subsystem of the processor-controlled device **350**. One or more processors communicate with the memory subsystem and execute either, some, or all applications. Because the processor-controlled device **350** is well known to those of ordinary skill in the art, no further explanation is needed.

[0065] FIG. **28** depicts other possible operating environments for additional aspects of the exemplary embodiments. FIG. **28** illustrates the entity's private software application **32** and/or the data layer application **132** operating within various other processor-controlled devices **350**. FIG. **28**, for example, illustrates that the entity's private software application **32** and/or the data layer application **132** may entirely or partially operate within a set-top box ("STB") (**352**), a personal/digital video recorder (PVR/DVR) **354**, a Global Positioning System (GPS) device **356**, an interactive television **358**, a tablet computer **360**, or any computer system, communications device, or processor-controlled device utilizing any of the processors above described and/or a digital signal processor (DP/DSP) **362**. Moreover, the processor-controlled device **350** may also include wearable devices (such as watches), radios, vehicle electronics, clocks, printers, gateways, mobile/implantable medical devices, and other apparatuses and systems. Because the architecture and operating principles of the various devices **350** are well known, the hardware and software componentry of the various devices **350** are not further shown and described.

[0066] Exemplary embodiments may be applied to any signaling standard. Most readers are thought familiar with the Global System for Mobile (GSM) communications signaling standard. Those of ordinary skill in the art, however, also recognize that exemplary embodiments are equally applicable to any communications device utilizing the Time Division Multiple Access signaling standard, the Code Division Multiple Access signaling standard, the "dual-mode" GSM-ANSI Interoperability Team (GAIT) signaling standard, or any variant of the GSM/CDMA/TDMA signaling standard. Exemplary embodiments may also be applied to other standards, such as the I.E.E.E. 802 family of standards, the Industrial, Scientific, and Medical band of the electromagnetic spectrum, BLUETOOTH and any other.

[0067] Exemplary embodiments may be physically embodied on or in a computer-readable storage medium. This computer-readable medium, for example, may include CD-ROM, DVD, tape, cassette, floppy disk, optical disk, memory card, memory drive, and large-capacity disks. This computer-readable medium, or media, could be distributed to end-subscribers, licensees, and assignees. A computer program product comprises processor-executable instructions for virtual execution of digital contracts, as the above paragraphs explain.

[0068] While the exemplary embodiments have been described with respect to various features, aspects, and embodiments, those skilled and unskilled in the art will recognize the exemplary embodiments are not so limited. Other variations, modifications, and alternative embodiments may be made without departing from the spirit and scope of the exemplary embodiments.

1. A method, comprising:
receiving, by a server, a private blockchain associated with a private entity, the private blockchain specifying a contract identifier that uniquely identifies a digital contract;
determining, by the server, a virtual machine that executes the digital contract;
instructing, by the server, the virtual machine to execute the digital contract; and
generating, by the server, a data record in a blockchain data layer that documents an execution of the digital contract by the virtual machine.

2. The method of claim 1, further comprising generating a cryptographic proof based on a hashing of the data record in the blockchain data layer.

3. The method of claim 2, further comprising publicly publishing the cryptographic proof via a public blockchain.

4. The method of claim 1, further comprising generating a cryptographic proof of the execution of the digital contract by the virtual machine.

5. The method of claim 5, further comprising publicly publishing the cryptographic proof via a public blockchain.

6. The method of claim 1, further comprising transacting a cryptocoinage in response to the execution of the digital contract.

7. The method of claim 1, further comprising transacting a cryptocoinage in response to generating the data record in the blockchain data layer.

8. A system, comprising:
a hardware processor; and
a memory device, the memory device storing instructions, the instructions when executed causing the hardware processor to perform operations, the operations comprising:
receiving a private blockchain associated with a private entity, the private blockchain specifying a contract identifier that uniquely identifies a digital contract;
querying an electronic database for the contract identifier, the electronic database electronically associating virtual machines to contract identifiers including the contract identifier specified by the private blockchain;
identifying the virtual machine in the electronic database that is electronically associated with the contract identifier specified by the private blockchain;
instructing the virtual machine to execute the digital contract; and
generating a data record in a blockchain data layer that documents an execution of the digital contract by the virtual machine.

9. The system of claim 8, wherein the operations further comprise generating a cryptographic proof based on a hashing of the data record in the blockchain data layer.

10. The system of claim 9, wherein the operations further comprise publicly publishing the cryptographic proof via a public blockchain.

11. The system of claim 8, wherein the operations further comprise generating a cryptographic proof of the execution of the digital contract by the virtual machine.

12. The system of claim 11, wherein the operations further comprise publicly publishing the cryptographic proof via a public blockchain.

13. The system of claim 8, wherein the operations further comprise transacting a cryptocoinage in response to the execution of the digital contract.

14. The system of claim 8, wherein the operations further comprise transacting a cryptocoinage in response to the generating of the data record in the blockchain data layer.

15. A memory device storing instructions that when executed cause a hardware processor to perform operations, the operations comprising:
receiving a private blockchain associated with a private entity, the private blockchain containing cryptographically hashed private data associated with private cryptocoinage issued by the private entity and specifying a contract identifier that uniquely identifies a digital contract;
querying an electronic database for the contract identifier, the electronic database electronically associating virtual machines to contract identifiers including the contract identifier specified by the private blockchain;
identifying the virtual machine in the electronic database that is electronically associated with the contract identifier specified by the private blockchain;
instructing the virtual machine to execute the digital contract;
generating a data record in a blockchain data layer that documents an execution of the digital contract by the virtual machine.
generating a cryptographic proof based on a hashing of the data record in the blockchain data layer;
publicly publishing the cryptographic proof via a public blockchain; and
identifying the data record in the blockchain data layer that is associated with the execution of the digital contract by the virtual machine.

16. The memory device of claim 15, wherein the operations further comprise receiving a request for the data record in the blockchain data layer that is associated with the execution of the digital contract by the virtual machine.

17. The memory device of claim 15, wherein the operations further comprise transacting a cryptocoinage in response to the execution of the digital contract by the virtual machine.

18. The memory device of claim 15, wherein the operations further comprise transacting a cryptocoinage in response to the generating of the data record in the blockchain data layer.

* * * * *