

What is claimed is:

1. In a transaction system where transactions are organized in blocks, wherein the transaction system comprises a plurality of workstations connected via a network, and each of the plurality of workstations executes a method of constructing~~a method of providing~~ new blocks B^r and B^{r+1} relative to a sequence of prior blocks $B^0, B^1, \dots, B^{r-1}$, the method comprising:

causing the block B^r to be constructed by one or more workstations of the plurality of workstations after confirming that the block B^r indicates initial states of transactions that are consistent with a state of the transactions in the block B^{r-1} ;

causing the block B^{r+1} to be constructed by one or more workstations of the plurality of workstations after confirming that the block B^{r+1} indicates initial states of transactions that are consistent with a state of the transactions in the block B^r ; and

causing a plurality of ~~entities~~-verifiers to verify previously- unverified transactions of the block B^r after the block B^{r+1} is constructed, wherein the plurality of verifiers comprises one or more workstations of the plurality of workstations.

2. A method, according to claim 1, wherein the transactions are smart contracts.

3. A method, according to claim 2, wherein ~~a user constructs~~ the block B^r is constructed in connection with posting a transaction to at least one of the smart contracts in the block B^r .

4. A method, according to claim 3, wherein ~~the user also posts~~ transaction results are posted for the smart contracts in the block B^r .

5. A method, according to claim 4, wherein ~~the user also posts~~ a number of steps for execution of the smart contracts results are posted for the smart contracts in the block B^r .

6. A method, according to claim 5, wherein ~~the user also posts~~ a fee is posted, wherein the fee is received by one or more of the plurality of verifiers ~~that the plurality of entities receive~~ for verifying transactions of the block B^r .

7. A method, according to claim 1, wherein the plurality of ~~entities~~-verifiers is a subset of all users of the block B^r .

8. A method, according to claim 7, wherein the plurality of ~~entities~~ verifiers is chosen randomly.

9. A method, according to claim 8, wherein choosing the plurality of verifiers~~entities~~ randomly includes applying a cryptographic hash function to data that includes at least one of: time information, information about one or more of the blocks, data included in one or more of the blocks, or data deduced from one or more of the blocks.

~~10. A method of causing a block B in a blockchain to be added to the blockchain, comprising:~~
~~causing an entity to receive information corresponding to a previous block;~~
~~causing the entity to receive a declarative invocation of a smart contract execution on a given input, wherein the declarative invocation declares relevant results of the execution and other relevant data;~~
~~causing the entity to verify syntactic validity of the invocation; and~~
~~causing the entity to incorporate the declarative invocation in the block B in response to verifying the syntactic validity of the invocation.~~

~~11. A method, according to claim 10, wherein the relevant results specify at least one of: net effects of the smart contract execution, a resulting state of the smart contract after execution, and a number of steps for execution and wherein the other relevant data specifies at least one of: a caller of the declarative invocation, time information, block information, and a payable fee.~~

~~12. A method, according to claim 11, wherein the declarative invocation is syntactically valid if the fee is adequate for the number of steps for execution and a payer of the fee has sufficient assets to pay the fee, and wherein the fee is paid if the declarative invocation appears in the blockchain.~~

~~13. A method of causing a set of verifiers in a blockchain to be assigned to verify a set S of one or more declarative invocations of smart contract executions on given inputs, wherein the declarative invocations declare relevant results of the smart contract executions, the method comprising:~~
~~the verifiers receiving the set S;~~
~~the verifiers determining which of the declarative invocations in S are semantically~~

valid; and

~~the verifiers providing authenticated responses of the one or more declarative invocations that are semantically valid.~~

~~14. A method, according to claim 13, wherein the set of verifiers are selected based on S, in a pseudo-random manner, via a given cryptographic function.~~

~~15. A method, according to claim 14, wherein at least one verifier determines that the at least one verifier has been selected via a computation that involves a secret key of the at least one verifier, and wherein the at least one verifier can prove to others that the at least one verifier has been selected.~~

~~16. A method, according to claim 13, wherein an invocation I in S is considered correct if responses of at least a given number of verifiers assigned to S indicate that I is semantically correct.~~

~~17. In a blockchain in which a set S of declarative smart contract invocations has already been registered, a set of verifiers assigned to S, and the verifiers have provided authenticated responses about which of the invocations of S are semantically valid, a method of causing a verdict of the invocations of S to be incorporated in the blockchain, comprising:~~

~~having an entity receive the authenticated responses of the verifiers;~~

~~having the entity deduce correctness of the invocations of S; and~~

~~having the entity incorporate in the blockchain a final verdict about correctness of the invocations of S.~~

~~18. A method, according to claim 17, wherein the final verdict of an invocation I in S indicates that I is correct if the responses of at least a given number of the verifiers assigned to S indicate that I is semantically valid.~~

~~19. A method, according to claim 17, wherein if the final verdict of an invocation I of S indicates that I is correct, causing other users to update the state of the smart contract and to consider the declared net effects of the execution of the smart contract to have taken place.~~

~~20. A method of causing a block B in a blockchain to be added to the blockchain, comprising:
causing an entity to receive information corresponding to a previous block;
causing receipt of a committed declarative invocation of a smart contract execution on
a given input, wherein the committed declarative invocation declares at least some relevant
results of the execution, a commitment to other relevant results of the execution, and other
relevant data;~~

~~causing verification of the syntactic validity of the committed declarative invocation;
and~~

~~causing incorporation of the committed declarative invocation in the block B in
response to verifying the syntactic validity of the committed declarative invocation.~~

~~21. A method, according to claim 20, wherein the relevant results specify at least one of: a net
effects of execution, a resulting state of the contract after execution, and a number of steps
and wherein the other relevant data specifies at least one of: a caller of the invocation, time
information, block information, and a payable fee.~~

~~22. A method, according to claim 21, wherein the declarative invocation is syntactically valid
if the fee is adequate for the number of steps and the payer of the fee has sufficient assets in
the system to pay the fee, and wherein the fee is paid if the invocation appears in the
blockchain.~~

~~23. A method of causing a set of verifiers in a blockchain to be assigned to verify a set S of
declarative invocations, including at least one committed declarative invocation I specifying
an execution e of a smart contract on given inputs, wherein I includes a commitment to at
least some relevant results rr of e, comprising:~~

~~the verifiers receiving the set S;~~

~~the verifiers reconstructing execution e so as to compute the relevant results rr; the
verifiers providing authenticated responses including personalized commitments to rr of the
verifiers.~~

~~24. A method, according to claim 23, wherein the assigned verifiers are selected based on S,
in a pseudo-random manner, via a given cryptographic function.~~

~~25. A method, according to claim 34, wherein at least one verifier can determine that the least one verifier has been selected via a computation that involves a secret key of the least one verifier, and wherein the least one verifier can prove to others that the least one verifier has been selected.~~

~~26. A method, according to claim 23, wherein the committed declarative invocation I in S is considered correct only if responses of at least a given number of verifiers assigned to S include a personalized commitment to the relevant results rr committed to in I.~~

~~27. In a blockchain in which a set S of declarative smart contract invocations has already been registered, a set of verifiers assigned to S, wherein the set S includes at least one committed declarative invocation I specifying an execution e of a smart contract on a given input, and wherein I includes a commitment to at least some relevant results rr of e, and the verifiers have provided authenticated responses about the invocations of S, a method of causing a verdict of the invocations of S to be incorporated in the blockchain, comprising:~~
~~having an entity receive the authenticated responses of the verifiers;~~
~~having the entity deduce correctness of the invocations of S; and~~
~~having the entity incorporate in the blockchain a final verdict about the correctness of the invocations of S.~~

~~28. A method, according to claim 27, wherein the responses of at least a given number of the verifiers assigned to S include a personalized commitment to the same relevant results rr committed to in I, and wherein the final verdict of the committed declarative invocation I in S indicates that I is correct.~~

~~29. A method, according to claim 28, wherein the final verdict of the correctness of I causes other users to update the state of the smart contract and to consider the net effects of the execution e of the smart contract to have taken place.~~

130. Computer software, provided in a non-transitory computer-readable medium, comprising:

executable code that causes a workstation in the transaction system to execute
~~implements~~ the method of one of the preceding claims 1-29.