Claims

1. A distributed network comprising a plurality of nodes, wherein each of the plurality of nodes is configured to run one or more computational units and each of the computational units comprises its own unit state; wherein the network is configured to

individually execute, by an execution subset of the plurality of nodes, a set of execution messages in a deterministic manner, thereby mutating the unit states of one or more of the computational units of the execution subset;

regularly make, by the nodes of the execution subset, a read snapshot of the unit states of the one or more computational units of the execution subset; and

provide, by one or more nodes of the execution subset, user access to the read snapshot;

regularly perform, by a consensus subset of the plurality of nodes, a consensus protocol configured to reach a consensus on a selection of execution messages from a currently available set of the set of execution messages;

wherein

the consensus protocol is configured to

receive and process the execution messages from the current set of execution messages;

generate a queue of input blocks from the execution messages according to a predefined consensus mechanism;

provide the queue of input blocks to a messaging protocol, wherein the messaging protocol is configured to process the input blocks and to schedule the execution messages of the input blocks for execution;

wherein the messaging protocol is configured to be clocked by the input blocks received from the consensus protocol;

each node of the execution subset comprises a state manager component configured to manage and store the read snapshots; and

the state manager component is configured to make the read snapshots at processing boundaries of the input blocks.

2. The network according to claim 1, wherein the network is configured to

execute, by a computational unit of one or more nodes of the execution subset, a computation on the read snapshot; and

provide, by the computational unit, a result of the computation to a user.

3. The network according to any of the preceding claims, wherein the network is configured to

receive, by one or more nodes of the execution subset, a read query from a user;

load, by the one or more nodes, the unit state of the corresponding computational unit associated with the read query;

run the computational unit with the read query; and

return, by the one or more nodes, a result of the read query, wherein the read query does not persistently change the unit state.

4. The network according to any of the preceding claims, wherein the network is further configured to

certify one or more parts of the read snapshot, thereby generating one or more certified parts of the read snapshot.

5. The network according to any of the preceding claims, wherein the network is configured to

regularly make, by the nodes of the execution subset, a recovery snapshot of the unit states of the one or more computational units of the execution subset, wherein the recovery snapshot is a snapshot that comprises all state of the nodes of the execution subset that is needed to restore a node;

certify, by a majority of the nodes of the execution subset, the recovery snapshot.

6. The network according to claim 5, wherein the network is configured to

store the recovery snapshot persistently at least until a subsequent recovery snapshot has been made.

7. The network according to claim 5 or 6, wherein the network is configured to perform the read snapshots at a first rate and the recovery snapshots at a second rate, wherein the first rate is higher than the second rate.

8. The network according to any of the preceding claims, wherein

each node of the consensus subset comprises a consensus component configured to perform the consensus protocol; and

each node of the execution subset comprises

an execution component configured to execute the selection of execution messages;

a messaging component configured to receive the selection of execution messages from the consensus component and to provide the selection of execution messages to the execution component;

a state manager component configured to manage and store the read snapshots;

a state reader component configured to handle read queries from users of the network; and/or

an ingress message handler component configured to handle mutating queries from users of the network.

9. The network according to any of the preceding claims, wherein the

network is configured to perform a plurality of processing loops in a sequential order with an increasing height index N, wherein N is an increasing integer, the plurality of processing loops being configured to

perform, at a first loop step, the consensus protocol;

individually execute, at a second loop step, the selection of execution messages;

make, at a third loop step, the read snapshot; and

provide, during the processing loop with the height index N, user access to the read snapshot made at the end of a previous height index, in particular the previous height index N-1.

10. The distributed network according to any of the preceding claims, wherein the consensus protocol is further configured to add one or more execution parameters to the input blocks, the execution parameters being selected from the group consisting of:

a random seed;

a designated execution time; and

a height index.

11. The distributed network according to any of the preceding claims, wherein the distributed network comprises a plurality of subnets;

each of the plurality of nodes is allocated to one of the plurality of subnets;

each of the plurality of nodes is configured to run a mainnet protocol client and a subnet protocol client, wherein

the subnet protocol client is configured to

regularly perform the consensus protocol;

individually execute the selection of execution messages;

regularly make the read snapshot; and

provide user access to the read snapshot; and

the mainnet protocol client is configured to distribute configuration data to the plurality of subnets.

12. A computer-implemented method for providing a user of a distributed network access to computational results computed by the distributed network, the distributed network comprising a plurality of nodes, the method comprising

running, on each of the plurality of nodes, one or more computational units, wherein each of the one or more computational units comprises its own unit state;

individually executing, by an execution subset of the plurality of nodes, a set of execution messages in a deterministic manner, thereby mutating the unit states of one or more of the computational units of the execution subset;

regularly making, by the nodes of the execution subset, a read snapshot of the unit states of the one or more computational units of the execution subset;

providing, by one or more nodes of the execution subset, user access to the read snapshot;

regularly performing, by a consensus subset of the plurality of nodes, a consensus protocol configured to reach a consensus on a selection of execution messages from a currently available set of the set of execution messages;

wherein

the consensus protocol is configured to

receive and process the execution messages from the current set of execution messages;

generate a queue of input blocks from the execution messages according to a predefined consensus mechanism;

provide the queue of input blocks to a messaging protocol, wherein the messaging protocol is configured to process the input blocks and to schedule the execution messages of the input blocks for execution;

wherein the messaging protocol is configured to be clocked by the input blocks received from the consensus protocol;

each node of the execution subset comprises a state manager component configured to manage and store the read snapshots; and

the state manager component is configured to make the read snapshots at processing boundaries of the input blocks.

13. A node for a distributed network,

wherein the node is configured to

run one or more computational units, wherein each of the computational units comprises its own unit state;

individually execute a set of execution messages in a deterministic manner, thereby mutating the unit states of one or more of the computational units;

regularly make a read snapshot of the unit states of the one or more computational units;

provide user access to the read snapshot;

regularly perform a consensus protocol configured to reach a consensus on a selection of execution messages from a currently available set of the set of execution messages;

wherein

the consensus protocol is configured to

receive and process the execution messages from the current set of execution messages;

generate a queue of input blocks from the execution messages according to a predefined consensus mechanism;

provide the queue of input blocks to a messaging protocol, wherein the messaging protocol is configured to process the input blocks and to schedule the execution messages of the input blocks for execution;

wherein the messaging protocol is configured to be clocked by the input blocks received from the consensus protocol;

wherein the node comprises a state manager component configured to manage and store the read snapshots; and

the state manager component is configured to make the read snapshots at processing boundaries of the input blocks.

14. A computer program product for operating a distributed network, the distributed network comprising a plurality of nodes, wherein each of the plurality of nodes is configured to run one or more computational units and each of the computational units comprises its own unit state; the computer program product comprising a computer readable storage medium having program instructions embodied therewith, the program instructions executable by one or more of the plurality of nodes to cause the one or more of the plurality of nodes to perform a method comprising

individually executing, by an execution subset of the plurality of nodes, a set of execution messages in a deterministic manner, thereby mutating the unit states of one or more of the computational units of the execution subset;

regularly making, by the nodes of the execution subset, a read snapshot of the unit states of the one or more computational units of the execution subset; and

providing, by one or more nodes of the execution subset, user access to the read snapshot;

regularly performing, by a consensus subset of the plurality of nodes, a consensus protocol configured to reach a consensus on a selection of execution messages from a currently available set of the set of execution messages;

wherein

the consensus protocol is configured to

receive and process the execution messages from the current set of execution messages;

generate a queue of input blocks from the execution messages according to a predefined consensus mechanism;

provide the queue of input blocks to a messaging protocol, wherein the messaging protocol is configured to process the input blocks and to schedule the execution messages of the input blocks for execution;

wherein the messaging protocol is configured to be clocked by the input blocks received from the consensus protocol;

each node of the execution subset comprises a state manager component configured to manage and store the read snapshots; and

the state manager component is configured to make the read snapshots at processing boundaries of the input blocks.

15. A computer program product for operating a node of a distributed network, the computer program product comprising a computer readable storage medium having program instructions embodied therewith, the program instructions executable by the node to cause the node to perform a method comprising

running one or more computational units, wherein each of the computational units comprises its own unit state;

individually executing a set of execution messages in a deterministic manner, thereby mutating the unit states of one or more of the computational units;

regularly making a read snapshot of the unit states of the one or more computational units; and

providing user access to the read snapshot;

regularly performing a consensus protocol configured to reach a consensus on a selection of execution messages from a currently available set of the set of execution messages;

wherein

the consensus protocol is configured to

receive and process the execution messages from the current set of execution messages;

generate a queue of input blocks from the execution messages according to a predefined consensus mechanism;

provide the queue of input blocks to a messaging protocol, wherein the messaging protocol is configured to process the input blocks and to schedule the execution messages of the input blocks for execution;

wherein the messaging protocol is configured to be clocked by the input blocks received from the consensus protocol;

wherein the node comprises a state manager component configured to manage and store the read snapshots; and

the state manager component is configured to make the read snapshots at processing boundaries of the input blocks.