

Claims

1. A method of operating a distributed database implemented in a layered network, the layered network comprising a core layer comprising one or more core nodes, one or more intermediate layers each comprising one or more intermediate layer nodes, and one or more outer layers each comprising one or more outer layer nodes; wherein each of the core nodes is a node of a blockchain network, at least some of the intermediate layer nodes are database nodes of the distributed database, and at least some of the outer layer nodes are client nodes of the distributed database, each database node storing at least part of the distributed database; the method comprising, at a first one of the database nodes:

receiving one or more update requests from one or more of the client nodes, each being a request to update a respective target entry in the distributed database; and

for each of the received update requests, determining whether the respective target entry is found in the part of the database stored at said first database node, and if so, making the update of the update request to the respective target entry in the part of the database stored at the first database node, and if not, forwarding the request to another of the database nodes which stores a part of the database including the respective target entry;

wherein at least one transaction including an indication of the one or more update requests is also recorded on a blockchain of the blockchain network.

2. The method of claim 1, wherein each client node has a connection within the layered network to at least one of the database nodes, and the first database node receives at least one of the update requests directly from a client node via one of the connections between client and database nodes.

3. The method of claim 1 or 2, wherein each of the database nodes has a connection within the layered network to at least one other of the database nodes, and one or both of:

at least one of the update requests is received indirectly via at least one other of the database nodes, other than the first database node, over at least one of the connections between database nodes, and/or

the respective target entry of at least one of the update requests is found in the part of the database stored in one of the other database nodes, and said forwarding comprises forwarding to the other database node via at least one of said connections between database nodes.

4. The method of any preceding claim, wherein said at least one transaction is formulated by at least one of the client nodes, and sent by the at least one client node to the core layer to be mined onto the blockchain.
5. The method of any preceding claims, wherein each client node has a connection within the layered network to at least one of the core nodes, enabling the client nodes to check whether update requests have been recorded to the blockchain.
6. The method of claim 5, wherein each client has a connection within the layered network to more than one of the core nodes.
7. The method of claim 6, wherein each client has a connection within the layered network to more than one but not all of the core nodes.
8. The method of any of claims 5 to 7, wherein said at least one transaction is sent by at least one of the client nodes directly to the core layer via at least one of said connections between client and core nodes.
9. The method of any of claims 1 to 3, wherein said at least one transaction is formulated by at least one other of the database nodes or intermediate layer nodes other than the first database node, and sent to the core layer by the at least one other node to be mined onto the blockchain.
10. The method of any preceding claim, wherein said at least one transaction is formulated and sent to the core layer by at least one of the client nodes or at least one other of the database nodes or other intermediate layer nodes other than the first database node; and the method further comprises, by the first database node:

inspecting the blockchain to check that the indication of the one or more update requests has been recorded on the blockchain, and/or inspecting a mempool of one or more miners to check that the at least one transaction has been accepted.

11. The method of claim 10, wherein said making of the update is conditional on said check.
12. The method of claim 10 or 11, wherein the first database node is configured to perform said inspection directly via a connection within the layered network between the first node and at least one of the core nodes.
13. The method of any of claims 1 to 3, wherein said at least one transaction is formulated by the first database node, and sent to the core layer by the first database node to be mined onto the blockchain.
14. The method of any preceding claim, wherein each of the database nodes has a connection within the layered network to at least one of the core nodes.
15. The method of claim 14, wherein each of the database nodes has a connection within the layered network to at least one but not all of the core nodes.
16. The method of claim 14 or 15, wherein the at least one transaction is sent to the core layer by at least one of the database nodes directly via at least one of the connections between database and core node.
17. The method of any preceding claim, wherein the indication of the one or more update requests included in the at least one transaction comprises: data content of the one or more updates, being the update to make to content of the respective data entry or entries.
18. The method of any preceding claim, wherein the indication of the one or more update requests included in the at least one transaction comprises: one or more hashes,

each hash being a hash of a preimage comprising data content of at least one of the updates.

19. The method of any preceding claim, wherein the indication of the one or more update requests included in the at least one transaction comprises: an ID of the database node or nodes storing the part of the database in which the respective target entry is found.

20. The method of any preceding claim, wherein: said one or more update requests are a plurality of update requests, the respective target entries of at least some of the update requests are the same entry in the database found in the part of the database stored at the first database node, and said making of the updates comprises making the updates according to a specified order.

21. The method of claim 20, wherein the indication of the one or more update requests included in the at least one transaction comprises: the specified order of the plurality of update requests.

22. The method of claim 21, wherein the at least one transaction originates from at least one of the client nodes or at least one other of the database nodes or other intermediate layer nodes other than the first database node, and wherein the method comprises:
the first database node reading the specified order from the blockchain or a mempool of a miner of the blockchain network, wherein said making of the updates comprises making the updates according to said order as determined from the blockchain or mempool.

23. The method of claim 20 or 21, wherein the order is determined by: at least one of the client nodes submitting to an ordering service, receiving back a specification of the order from the ordering service, and including the specified order in at least one of the update requests.

24. The method of claim 20 or 21, wherein the specified order is determined by an ordering service implemented at the first database node.

25. The method of any preceding claim, wherein one or more entries of the database are replicated across the parts of the database stored on a plurality of the database nodes.

26. The method of claim 25, wherein the target entry of at least one of the update requests is found in both the part of the database stored on the first database node and another of the database nodes, such that the method comprises both making the update at the first database node and forwarding the request to the other database node.

27. The method of any preceding claim, wherein the target entry of at least one of the update requests is found in part of the database stored on another of the database nodes other than the first database node, and said forwarding comprises:

determining whether the other database node is currently reachable over the layered network, and forwarding the update request to the other database node on condition that it is determined to be reachable according to said determination.

28. The method of claim 27, wherein the other database node is temporarily disconnected from the layered network, and said indication in the at least one transaction enables said other database node to make the update of the at least one update request when that other database node reconnects to the layered network.

29. The method of any preceding claim, wherein for a period the first database node become temporarily disconnected from the layered network, and the method further comprises, when the first database nodes becomes reconnected to the layered network:

the first node inspecting the blockchain, or a mempool of a miner of the blockchain network, to check for one or more further transactions recording any update request received by one or more other database nodes, while the first database node was disconnected, relating to a respective target entry in a part of the database stored by said first database node, and making any such update.

30. The method of any of claims 10 to 12 or claim 29, wherein the first node is configured to perform said inspection directly via a connection within the layered network between the first node and at least one of the core nodes.
31. The method of any preceding claim, wherein the core layer is complete.
32. The method of any preceding claim, wherein the layered network as a whole is non-complete.
33. The method of any preceding claim, wherein each of the client nodes and/or database nodes is certified by a certificate authority.
34. Computer equipment comprising:
memory comprising one or more memory units,
processing apparatus comprising one or more processing units, and
a network interface comprising one or more network interface units;
wherein the memory stores code arranged to run on the processing apparatus, the code being configured so as when run on the processing apparatus to operate the computer equipment as said first database node by performing the method of any preceding claim, including receiving the update requests via the network interface.
35. A computer program embodied on computer readable storage and configured so as when run on one or more processors to perform the method of any of claims 1 to 33.
36. A method of using a distributed database a method of operating a distributed database implemented in a layered network, the layered network comprising a core layer comprising one or more core nodes, one or more intermediate layers each comprising one or more intermediate layer nodes, and one or more outer layers each comprising one or more outer layer nodes; wherein each of the core nodes is a node of a blockchain network, at least some of the intermediate layer nodes are database nodes of the distributed database, and at least some of the outer layer nodes are client nodes of the distributed

database, each database node storing at least part of the distributed database; the method comprising, at a first one of the client nodes:

 sending one or more update requests to one or more of the database nodes, each being a request to update a respective target entry in the distributed database; and

 using a connection within the layered network between the first client node and at least one of the core nodes in the core layer to directly:

 I) send at least one transaction to be recorded on a blockchain of the blockchain network, the at least one transaction including an indication of the one or more update requests is also, and/or

 II) check that at least one transaction including an indication of the one or more update requests has been recorded on the blockchain or at least accepted into a mempool of a miner of the blockchain network.

37. The method of claim 36, wherein the first client node is configured to communicate with the at least one core node, including to send said at least one transaction and/or perform said check, using a communication protocol in which messages take the form of:

 a) a transaction sent from client node to core node;

 b) a query from client node to core node as to whether a transaction has been accepted into a miner's mempool, and a corresponding response from the core node;

 c) a request from client node to core node for a Merkle proof that a transaction has been mined into a block, and a response from the core node comprising the Merkle proof; and/or

 d) a request from client node to core node for a list of block headers, and a response from the core node comprising the list of block headers.

38. Computer equipment comprising:

 memory comprising one or more memory units,

 processing apparatus comprising one or more processing units, and

 a network interface comprising one or more network interface units;

 wherein the memory stores code arranged to run on the processing apparatus, the code being configured so as when run on the processing apparatus to operate the computer

equipment as said first client node by performing the method of claim 36 or 37, including conducting I) and/or II) via the network interface.

39. A computer program embodied on computer readable storage and configured so as when run on one or more processors to perform the method of claim 36 or 37.