

CLAIMS

1. A computer-implemented method for updating databases that store blockchain data, the method comprising:

identifying (602), based on a transaction associated with a blockchain network, an account identifier (ID) of a blockchain account involved in the transaction, wherein the transaction is included in a current block to be appended to a blockchain associated with the blockchain network;

appending the current block to the blockchain by performing a consensus algorithm;

updating (604), in a current state object database, an account state corresponding to the account ID of the blockchain account to an updated account state based on the blockchain transaction after the current block is appended to the blockchain, wherein the current state object database is a key-value database that stores key-value pairs of account states mapped to account IDs of blockchain account;

wherein the method is characterised in that upon determining a need, updating, in the current state object database, the updated account state based on additional account information of the blockchain account to a second updated account state, wherein the additional account information is loaded from a separate database having a lower storage cost than the current state object database, wherein the additional account information of the blockchain account comprises infrequently accessed account data of the blockchain account;

hashing (606) the second updated account state to generate a second hash value of the second updated account state;

identifying (608), in a current state database, a first hash value of the account state corresponding to the blockchain account based on the account ID of the blockchain account, wherein the current state database stores a state tree that is configured to be traversed for current state data, wherein nodes of the state tree store key-value pairs that comprise account IDs and corresponding hash values of the account states of the corresponding blockchain accounts; and

updating (610), in the current state database, the first hash value of the account state to the second hash value of the second updated account state.

2. The method of claim 1, wherein the current state object database stores, for each blockchain account of the blockchain network, a mapping between an account ID and an account state of the corresponding blockchain account.

3. The method of claim 1, wherein the consensus algorithm is based on one of proof of work (PoW), proof of stake (PoS), and practical Byzantine fault tolerance (PBFT).
4. The method of any one of the previous claims, wherein the state tree is a location-addressed state tree.
5. The method of claim 4, wherein the location-addressed state tree is a current state tree of a fixed-depth Merkle tree (FDMT) or a bucket tree.
6. The method of any one of the previous claims, wherein the account ID identifies a relative location of a branch node or a leaf node at a corresponding layer of the state tree for locating the hash value of the account state.
7. The method of claim 6, wherein the hash value of the account state corresponding to the blockchain account is a first hash value, the leaf node stores a plurality of hash values of account states associated with a plurality of blockchain accounts, and the account ID further identifies a relative location of the first hash value among the hash values stored in the leaf node.
8. The method of any previous claim, wherein the consensus algorithm is performed after one or more of existence, validity, and authenticity of the transaction are verified.
9. A computer-implemented system for updating databases that store blockchain data, comprising:
 - one or more processors; and
 - one or more computer-readable memories coupled to the one or more processors and having instructions stored thereon that are executable by the one or more processors to perform the method of any of claims 1 to 8.
10. An apparatus for updating databases that store blockchain data, the apparatus comprising a plurality of modules for performing the method of any of claims 1 to 8.