CLAIMS

1. A data processing method, performed by a blockchain node in a target blockchain based on a node internal memory, comprising:

acquiring a transaction in a first block, and acquiring a data reading contract, which a contract for reading data of the transaction, from a chain reading cache of the node internal memory, the first block being a block to be written into a bifurcated blockchain in the node internal memory;

invoking the data reading contract to acquire a write cache mapping table, and determining target read data based on a first block cache of the first block in the write cache mapping table; and

executing the transaction based on the target read data, storing target cached data, which is obtained as a result of executing of the transaction, to the first block cache, and writing the first block into the bifurcated blockchain.

2. The method according to claim 1, wherein the chain reading cache is established for the target blockchain in a cache space of the node internal memory, and the chain reading cache is configured to temporarily store hot-spot data that is frequently accessed; the hot-spot data comprising the data reading contract.

3. The method according to claim 1, further comprising:

determining, as a target block, a block which, in the node internal memory, belongs to a longest chain on the bifurcated blockchain, and writing data in a target block cache of the target block into a blockchain database corresponding to the target blockchain; a block generation timestamp of the target block being earlier than a block generation timestamp of the first block.

4. The method according to claim 1, wherein the acquiring the transaction and the acquiring the data reading contract comprises:

acquiring the first block containing the transaction, and generating, during performing consensus processing on the first block, a transaction processing instruction containing a hot-spot reading auxiliary parameter; the hot-spot reading auxiliary parameter comprising: a contract invoking address based on which a transaction contract of the transaction is executed;

extracting the contract invoking address from the transaction processing instruction, and searching, based on the contract invoking address, a hot-spot contract matched with a contract name mapped to the contract invoking address from the chain reading cache of the node internal memory; the hot-spot contract being a transaction contract which is determined in the chain reading cache and has a cumulative access frequency greater than an access threshold; and

acquiring, as the data reading contract, the hot-spot contract matched with the contract name.

5. The method according to claim 1, wherein the transaction comprises: a target transaction for asset transferring initiated from a first user to a second user; a contract function in the data reading contract comprising a first execution function and a second execution function; the first execution function configured to read first asset data in a user account address of the first user; the second execution function configured to read a user account address of the second user;

the invoking the data reading contract and the determining the target read data comprises:

acquiring the write cache mapping table based on the data reading contract being run in a virtual machine; acquiring, as the first block cache in the write cache mapping table, a write cache mapped to a first block hash value of the first block; and acquiring a historical execution result of a historical transaction stored in the first block cache; a transaction execution order of the historical transaction being earlier than a transaction execution order of the target transaction;

searching the historical execution result of the first block cache for the user account address of the first user based on the first execution function in the data reading contract, and searching the historical execution result of the first block cache for the user account address of the second user based on the second execution function in the data reading contract;

reading the first asset data of the first user from the user account address of the first user based on the user account address of the first user being found in the first block cache; and

determining the read first asset data and the user account address of the second user to be asset reading data of the target transaction based on the user account address of the second user being found in the first block cache, and determining the target read data based on the asset reading data.

6. The method according to claim 5, wherein the bifurcated blockchain contains a second block; the second block being a previous block of the first block; the write cache mapping table containing a second block hash value of the second block;

the method further comprises:

based on the user account address of the first user being not found in the first block cache, determining a block header of the first block in the bifurcated blockchain based on the first block hash value mapped to the first block cache , and acquiring a father block hash value of the first block as the second block hash value of the second block in the block header of the first block;

acquiring the write cache mapped to the second block hash value as a second block cache in the write cache mapping relationship table; a logical query level of the second block cache being

lower than a logical query level of the first block cache;

querying the user account address of the first user in the second block cache; and

reading the first asset data of the first user from the user account address of the first user based on the user account address of the first user being queried in the second block cache.

7. The method according to claim 6, wherein the bifurcated blockchain contains a first branch chain, the first branch chain containing a third block and a fourth block, the third block being a previous block of the second block, and a block generation timestamp of the fourth block being earlier than a block generation timestamp of the third block;

the write cache mapping table contains a third block hash value of the third block and a fourth block hash value of the fourth block, a logical query level of a third block cache mapped to the third block hash value being lower than the logical query level of the second block cache, and a logical query level of a fourth block cache mapped to the fourth block hash value being lower than the logical query level of the third block cache; and

the method further comprises:

querying the user account address of the first user in a blockchain database corresponding to the target blockchain based on the user account address of the first user being not found in the second block cache, the third block cache, the fourth block cache; and

reading the first asset data of the first user from the user account address of the first user based on the user account address of the first user being queried in the blockchain database.

8. The method according to claim 3, wherein the acquiring the target block, and writing the data in the target block cache of the target block into the blockchain database comprises:

acquiring a longest chain selection policy in the bifurcated blockchain, screening a block that satisfies the longest chain selection policy in the bifurcated blockchain based on a block height of the first block, and acquiring the block that satisfies the longest chain selection policy as the target block, the target block being a block cached in the node internal memory and historically written into the bifurcated blockchain; and

writing the target block into the target blockchain based on a determination, in the node internal memory, that the target block belongs to the longest chain on the bifurcated blockchain, and writing the data in the target block cache of the target block into the blockchain database corresponding to the target blockchain.

9. The method according to claim 8, wherein the chain reading cache contains historical chain

data comprising data that is successfully written into a historical block cache of the target blockchain at a historical time, the historical block cache being a write cache of a historical block in the bifurcated blockchain, and a block generation timestamp of the historical block being earlier than a block generation timestamp of the target block;

the method further comprises:

acquiring the data in the target block cache as target chain data on the target blockchain, writing the target chain data into the chain reading cache, and updating the historical chain data in the chain reading cache with the target chain data.

10. The method according to claim 9, wherein a bifurcated block with a same block height as that of the target block exists on the bifurcated blockchain;

the method further comprises:

determining a target block height of the target block in the bifurcated blockchain, acquiring a hash relationship table from the node internal memory, and determining a target list column where the target block height is located in the hash relationship table;

acquiring a bifurcated block hash value of the bifurcated block from the target list column, and determining a bifurcated block cache corresponding to the bifurcated block hash value in the write cache mapping relationship table; and

clearing away the target chain data in the target block cache, and clearing away data in the bifurcated block cache.

11. The method according to claim 1, further comprising:

acquiring the data reading contract from a blockchain database associated with the node internal memory based on the data reading contract being not acquired in the chain reading cache.

12. The method according to claim 3, further comprising:

writing the target cached data in the first block cache into the blockchain database corresponding to the target blockchain based on a determination, in the node internal memory, that the first block belongs to the longest chain on the bifurcated blockchain.

13. The method according to claim 12, wherein the transaction comprises a first transaction and a second transaction, the target cached data comprises first cached data corresponding to the first transaction and second cached data corresponding to the second transaction; the second transaction being a previous transaction of the first transaction;

the writing the target cached data in the first block cache into the blockchain database corresponding to the target blockchain comprises:

writing the first cached data in the first block cache into the blockchain database corresponding to the target blockchain, and writing the second cached data in the first block cache into the blockchain database.

14. The method according to claim 1, further comprising:

determining a block that satisfies a cache duration condition in the bifurcated blockchain to be a block to be destroyed based on monitoring that a cache capacity of the node internal memory reaches a cache threshold; and

performing block destruction processing on the block to be destroyed in the bifurcated blockchain.

15. A data processing apparatus based on a node internal memory in a target blockchain, the apparatus being deployed on a blockchain node, comprising:

at least one memory configured to store program code; and

at least one processor configured to read the program code and operate as instructed by the program code, the program code including:

transaction acquisition code configured to cause the at least one processor to acquire a transaction in a first block, and acquire a data reading contract, which is a contract for reading data of the transaction, from a chain reading cache of the node internal memory; the first block being a block to be written into a bifurcated blockchain in the node internal memory;

contract invoking code configured to cause the at least one processor to invoke the data reading contract to acquire a write cache mapping table, and determine target read data based on a first block cache of the first block in the write cache mapping table; and

transaction execution code configured to cause the at least one processor to execute the transaction based on the target read data, store target cached data, which is obtained as a result of executing of the transaction, to the first block cache, and write the first block into the bifurcated blockchain.

16. The apparatus according to claim 15, further comprising:

longest chain determination code configured to cause the at least one processor to determine, as a target block, a block that, in the node internal memory, belongs to a longest chain on the bifurcated blockchain, and write data in a target block cache of the target block into a blockchain

database corresponding to the target blockchain; a block generation timestamp of the target block being greater than the block generation timestamp of the first block.

17. A computer device, comprising: a processor and a memory,

the processor being connected to the memory, the memory being configured to store a computer program, and the processor being configured to invoke the computer program, to cause the computer program to perform the method according to any one of claims 1 to 14.

18. A non-transitory computer-readable storage medium, storing a computer program, the computer program being loaded and executed by a processor to cause a computer device with the processor to perform the method according to any one of claims 1 to 14.

19. A computer program product, the computer program product, when executed, being configured to perform the method according to any one of claims 1 to 14.