What is claimed is:

1. A method for archiving block data of a blockchain, comprising:

detecting whether there are block data which satisfy an archiving condition in a node of a blockchain, wherein the archiving condition comprises a height value of the block data and an access frequency of the block data;

when the block data which satisfy the archiving condition are detected, locally copying the block data which satisfy the archiving condition, compressing the block data to generate a compressed package file, and uploading the compressed package file to a reliable distributed storage system;

when the compressed package file is successfully uploaded to the reliable distributed storage system, acquiring a hash value of the compressed package file, and inserting the hash value of the compressed package file and a record of a height range of the block data in the compressed package file into an archived file hash table.

2. The method of claim 1, wherein detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain comprises:

checking the number of the block data locally stored in the node of the blockchain;

when the number of the block data locally stored is larger than a preset number, acquiring the preset number of the block data from the block data with the lowest height value; and

when the access frequency of each of the preset number of the block data is less than a preset access frequency, determining that the preset number of the block data are the block data which satisfy the archiving condition.

3. The method of claim 2, wherein when the access frequency of each of the preset number of the block data is less than the preset access frequency, determining that the preset number of the block data are the block data which satisfy the archiving condition comprises:

recording the block data, the access frequency of which is higher than the preset access frequency in a preset period, and the corresponding height value ;

matching the height values of the preset number of the block data with the height values of the block data, the access frequency of which is higher than the preset access frequency;

when none of the height values of the preset number of the block data match the height values of the block data, the access frequency of which is higher than the preset access frequency, determining that the preset number of the block data are the block data which satisfy the archiving condition.

4. The method of claim 2 or claim 3, wherein detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain comprises:

periodically detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain, wherein a period of the detecting step is less than a time for generating the preset number of the block data.

5. The method of any one of claims 1 to 4, further comprising:

maintaining, in a database of the node of the blockchain, the archived file hash table, wherein the archived file hash table is adapted to save a corresponding relationship between the hash value of the archived compressed package file in the node of the blockchain and the height range of the block data in the compressed package file.

6. The method of any one of claims 1 to 5, further comprising:

when the compressed package file is not successfully uploaded to the reliable distributed storage system, outputting an error log and re-uploading the compressed package file to the reliable distributed storage system;

when the compressed package file is not successfully re-uploaded to the reliable distributed storage system, returning to the step that detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain; and

deleting redundant data generated.

7. A method for querying block data of a blockchain, comprising:

acquiring a query request, wherein the query request comprises a height value of block data to be queried;

when the block data to be queried are archived block data, querying, in an archived file hash table, a hash value of a compressed package file corresponding to the height value, according to the height value, wherein the archived file hash table comprises a corresponding relationship between the hash value of the compressed package file of the archived block data and a height range of the block data in the compressed package file;

when the hash value of the compressed package file corresponding to the height value is queried, determining, under a locally downloaded historical data directory, whether there is a folder named with the same hash value;

when the folder named with the same hash value is queried, accessing, under the historical data directory, the block data under the folder named with the same hash value for querying;

when the folder named with the same hash value is not queried, downloading the compressed package file corresponding to the hash value from a reliable distributed storage system, decompressing the compressed package file, and returning a query result.

8. The method of claim 7, wherein when the hash value of the compressed package file corresponding to the height value is not queried, the query request is forwarded to other nodes.

9. The method of claim 7 or claim 8, wherein after the query request is acquired, the method further comprises:

determining, according to whether the height value less than the lowest height value of the block data stored locally in the node of the blockchain, whether the block data to be queried have been archived.

10. A device for archiving block data of a blockchain, comprising a memory and a processor, wherein the processor executes the following steps:

detecting whether there are block data which satisfy an archiving condition in a node of a blockchain, the archiving condition comprising a height value of the block data and an access frequency of the block data;

when the block data which satisfy the archiving condition are detected, locally copying the block data which satisfy the archiving condition, compressing the block data to generate a compressed package file, and uploading the compressed package file to a reliable distributed storage system;

when the compressed package file is successfully uploaded to the reliable distributed storage system, acquiring a hash value of the compressed package file and inserting the hash value of the compressed package file and the record of a height range of the block data in the compressed package file into an archived file hash table.

11. The device of claim 10, wherein when executing the step of detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain, the processor specifically executes the following steps:

checking the number of the block data locally stored in the node of the blockchain;

when the number of the block data locally stored is larger than a preset number, acquiring the preset number of the block data from the block data with the lowest height value; and

when the access frequency of each of the preset number of the block data is less than a preset access frequency, determining that the preset number of the block data are the block data which satisfy the archiving condition.

12. The device of claim 11, wherein when executing the step of when the access frequency of each of the preset number of the block data is less than the preset access frequency, determining that the preset number of the block data are the block data which satisfy the archiving condition, the processor specifically executes the following steps:

recording the block data, the access frequency of which is higher than the preset access frequency in a preset period, and the corresponding height value;

matching the height values of the preset number of the block data with the height values of the block data, the access frequency of which is higher than the preset access frequency; and

when none of the height values of the preset number of the block data match the height values of the block data, the access frequency of which is higher than the preset access frequency, determining that the preset number of the block data are the block data which satisfy the archiving condition.

13. The device of any one of claims 10 to 12, wherein when executing the step of detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain, the processor specifically executes the following steps:

periodically detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain, a period of which is less than a time for generating the preset number of the block data.

14. The device of any one of claims 10 to 13, wherein the processor further executes the following step:

maintaining the archived file hash table in a database of the node of the blockchain, the archived file hash table adapted to save a corresponding relationship between a hash value of the archived compressed package file in the node of the blockchain and the height range of the block data in the compressed package file.

15. The device of any one of claims 10 to 14, wherein the processor further executes the following steps:

when the compressed package file is not successfully uploaded to the reliable distributed storage system, outputting an error log and re-upload the compressed package file to the reliable distributed storage system;

when the compressed package file is failed in re-uploading to the reliable distributed storage system, returning to the step that detecting whether there are the block data which satisfy the archiving condition in the node of the blockchain; and

deleting redundant data generated.

16. A device for querying block data of a blockchain, comprising a memory and a processor, wherein, the processor executes the following steps:

acquiring a query request, the query request comprising a height value of block data to be queried;

when the block data to be queried are archived block data, querying, in an archived file hash table, a hash value of a compressed package file corresponding to the height value, according to the height value, the archived file hash table comprising a corresponding relationship between the hash value of the compressed package file of the archived block data and a height range of the block data in the compressed package file;

when the hash value of the compressed package file corresponding to the height value is queried, determining, under a locally downloaded historical data directory, whether there is a folder named with the same hash value;

when the folder named with the same hash value is queried, accessing under the historical data directory, the block data under the folder named with the same hash value for querying;

when the folder named with same the hash value is not queried, downloading the compressed package file corresponding to the hash value from a reliable distributed storage system, decompressing the compressed package file, and returning a query result.

17. The device of claim 16, wherein when the hash value of the compressed package file corresponding to the height value is not queried, the query request is forwarded to other nodes.

18. The device of claim 16 or claim 17, wherein after the query request is acquired, wherein the processor further executes the following step:

determining, according to whether the height value is less than the lowest height value of the block data locally stored in the node of the blockchain, whether the block data to be queried have been archived.

19. A computer device, comprising a memory, a processor, and a computer program stored on the memory for execution by the processor, wherein, when the computer program is executed by the processor, the method for archiving block data of blockchain, as claimed in any one of claims 1 to 6, is implemented, and/or steps of the method for querying block data of blockchain, as claimed in any one of claims 7 to 9, is implemented.

20. A computer readable storage medium, on which a computer program is stored, wherein when the computer program is executed by a processor, and the method for archiving block data of blockchain, as claimed in any one of claims 1 to 6, is implemented, and/or steps of the method for querying block data of blockchain, as claimed in any one of claims 7 to 9, is implemented.