CLAIMS

1. A method for executing a smart contract, comprising:

receiving, by a blockchain node of a blockchain network, a request for creating the smart contract, the request including bytecodes of the smart contract;

starting, by the blockchain node of the blockchain network after completing deployment of the smart contract of bytecodes in the blockchain network, to compile a bytecode of the bytecodes of the smart contract into a machine code by using ahead of time (AoT) compilation within a relatively idle time period with respect to service load of the blockchain node and storing the machine code, wherein the deployment comprises that the blockchain node can locally execute the transaction for creating the smart contract and create a corresponding contract account and a corresponding contract address; and

in execution of the smart contract, executing, by the blockchain node, the machine code of the smart contract in response to the machine code being locally stored.

2. The method according to claim 1, wherein the executing, by the blockchain node, of the machine code includes:

executing, by the blockchain node, a stored machine code corresponding to an invoked function in the smart contract.

3. The method according to claim 1, comprising:

interpreting and executing, by the blockchain node in executing the deployed smart contract, a bytecode of the bytecodes of the smart contract in response to a machine code of the smart contract not being locally stored.

4. The method according to claim 3, wherein the interpreting and executing, by the blockchain node, of the bytecode of the smart contract includes:

interpreting and executing, by the blockchain node, a bytecode corresponding to an invoked function in the smart contract.

5. The method according to claim 3, wherein the interpreting and executing, by the blockchain node in executing the deployed smart contract, the bytecode of the smart contract in response to the machine code of the smart contract not being locally stored includes:

querying, by the blockchain node in executing the deployed smart contract, a current AoT compilation status of the bytecode of the smart contract in response to the machine code of the smart contract not being locally stored;

in response to the bytecode being in AoT compilation, interpreting and executing the bytecode of the smart contract; and

in response to the bytecode not being in AoT compilation, starting AoT compilation to interpret and execute the bytecode of the smart contract.

6. The method according to claim 1, comprising:

performing just-in-time (JIT) compilation on and executing, by the blockchain node in executing the deployed smart contract, a bytecode of the smart contract in response to a machine code of the smart contract not being locally stored.

7. The method according to claim 6, wherein the performing JIT compilation on the bytecode of the smart contract includes:

interpreting the bytecode of the smart contract statement by statement, and caching the interpreted machine code for subsequent invoking; or

in a process of one or more of interpreting or executing the bytecode of the smart contract, identifying a machine code of a hotspot, and compiling and caching the machine code of the hotspot for subsequent invoking.

8. The method according to claim 7, wherein a granularity level of the machine code of the hotspot includes a function level or a code block level.

9. The method according to claim 7, wherein the performing JIT compilation on the bytecode of the smart contract further includes: performing compilation optimization in a process of performing JIT interpretation on the bytecode of the smart contract.

10. The method according to any one of claims 1, 2, 3, 5, or 6, wherein the storing is caching.

11. A blockchain node for executing a smart contract, comprising:

one or more processors; and

one or more computer-readable memories coupled to the one or more processors and having executable instructions stored thereon that are executable by the one or more processors to perform acts including:

receiving a request for creating the smart contract, the request including bytecodes of the smart contract;

starting, after completing deployment of the smart contract of bytecodes in a blockchain network, to compile a bytecode of the bytecodes of the smart contract into a machine code by using ahead of time compilation and storing the machine code within a relatively idle time period with respect to service load of the blockchain node and storing the machine code, wherein the deployment comprises that the blockchain node can locally execute the transaction for creating the smart contract and create a corresponding contract account and a corresponding contract address; and

in execution of the smart contract, executing the machine code of the smart contract in response to the machine code being locally stored.

12. The blockchain node according to claim 11, wherein the acts include:

interpreting and executing, in executing the deployed smart contract, a bytecode of the bytecodes of the smart contract in response to a machine code of the smart contract not being locally stored.

13. The blockchain node according to claim 12, wherein the interpreting and executing the bytecode of the deployed smart contract in response to the machine code of the smart contract not being locally stored includes:

querying, in executing the deployed smart contract, a current AoT compilation status of the bytecode of the smart contract in response to the machine code of the smart contract not being locally stored;

in response to the bytecode being in AoT compilation, interpreting and executing the bytecode of the smart contract; and

in response to the bytecode not being in AoT compilation, starting AoT compilation to interpret and execute the bytecode of the smart contract.

14. The blockchain node according to claim 11, wherein the acts include:

performing just-in-time compilation on and executing, in executing the deployed smart contract, a bytecode of the smart contract in response to a machine code of the smart contract not being locally stored.

15. The blockchain node according to claim 14, wherein the performing JIT compilation on the bytecode of the smart contract includes at least one of:

interpreting the bytecode of the smart contract statement by statement, and caching the interpreted machine code for subsequent invoking; or

in a process of one or more of interpreting or executing the bytecode of the smart contract, identifying a machine code of a hotspot, and compiling and caching the machine code of the hotspot for subsequent invoking.

16. A non-transitory storage medium, the storage medium having executable instructions stored thereon that are executable by one or more processors to perform acts comprising:

receiving, by a blockchain node, a request for creating the smart contract, the request including bytecodes of the smart contract;

starting, after completing deployment of the smart contract of bytecodes in a blockchain network, to compile a bytecode of the bytecodes of the smart contract into a machine code by using ahead of time compilation and storing the machine code within a relatively idle time period with respect to service load of the blockchain node and storing the machine code, wherein the deployment comprises that the blockchain node can locally execute the transaction for creating the smart contract and create a corresponding contract account and a corresponding contract address; and

in execution of the smart contract, executing the machine code of the smart contract in response to the machine code being locally stored.

17. The storage medium according to claim 16, wherein the acts include:

performing just-in-time compilation on and executing, in executing the deployed smart contract, a bytecode of the smart contract in response to a machine code of the smart contract not being locally stored.