CLAIMS:

1. A computer-implemented method for implementing blockchain contracts, comprising:

obtaining bytecode of a blockchain contract, wherein the bytecode comprises an indicator indicating a type of the blockchain contract, the type corresponding to a programming language in which the blockchain contract is written;

determining a virtual machine corresponding to the type based at least on the indicator; and

triggering the determined virtual machine to execute the blockchain contract.

2. The method of claim 1, wherein obtaining the bytecode of the blockchain contract comprises:

obtaining from a client device the bytecode of the blockchain contract, wherein the bytecode is compiled from source code of the blockchain contract, and the source code comprises the indicator.

3. The method of claim 1, wherein obtaining the bytecode of the blockchain contract comprises:

obtaining from a client device source code of the blockchain contract, the source code comprising the indicator; and

compiling the source code to obtain the bytecode of the blockchain contract.

4. The method of claim 1, wherein obtaining the bytecode of the blockchain contract comprises:

obtaining from a client device raw bytecode for executing the blockchain contract, wherein the raw bytecode is compiled from raw source code for executing the blockchain contract; and

adding the indicator to the raw bytecode to obtain the bytecode of the blockchain contract.

5. The method of claim 1, wherein obtaining the bytecode of the blockchain contract comprises:

obtaining from a client device raw source code for executing the blockchain contract;

compiling the raw source code to obtain raw bytecode; and

adding the indicator to the raw bytecode to obtain the bytecode of the blockchain contract.

6. The method of any preceding claim, wherein:

the indicator corresponds to a first byte of the bytecode of the blockchain contract.

7. The method of any preceding claim, wherein:

the type of the blockchain contract is solidity contract or native contract.

8. The method of any preceding claim, wherein:

the programming language comprises c++, java, solidity, or python;

the virtual machine corresponding to c++ is GNU compiler collection (GCC);

the virtual machine corresponding to java is java virtual machine (JVM);

the virtual machine corresponding to solidity is ethereum virtual machine (EVM); and

the virtual machine corresponding to python is python virtual machine (PVM).

9. A system for implementing blockchain contracts, 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 implementing blockchain contracts, comprising a plurality of modules for performing the method of any of claims 1 to 8.

11. A computer-implemented method for implementing blockchain contracts, comprising:

generating bytecode of a blockchain contract, wherein the bytecode comprises an indicator indicating a type of the blockchain contract, the type corresponding to a programming language in which the blockchain contract is written; and

transmitting the bytecode of the blockchain contract to a blockchain node for the blockchain node to trigger a virtual machine corresponding to the type based on the indicator to execute the blockchain contract.

12. The method of claim 11, wherein generating the bytecode of the blockchain contract comprises:

compiling source code of the blockchain contract to obtain the bytecode of the blockchain, the source code comprising the indicator.

13. The method of claim 11, wherein generating the bytecode of the blockchain contract comprises:

compiling raw source code for executing the blockchain contract to obtain raw bytecode for executing the blockchain contract; and

adding the indicator to the raw bytecode to obtain the bytecode of the blockchain contract.

14. The method of any of claims 11-13, wherein:

the indicator corresponds to a first byte of the bytecode of the blockchain contract.

15. The method of any of claims 11-13, wherein:

the type of the blockchain contract is solidity contract or native contract.

16. The method of any of claims 11-13, wherein:

the programming language comprises c++, java, solidity, or python;

the virtual machine corresponding to c++ is GNU compiler collection (GCC);

the virtual machine corresponding to java is java virtual machine (JVM);

the virtual machine corresponding to solidity is ethereum virtual machine (EVM); and

the virtual machine corresponding to python is python virtual machine (PVM).

17. A system for implementing blockchain contracts, 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 11 to 16.

18. An apparatus for implementing blockchain contracts, comprising a plurality of modules for performing the method of any of claims 11 to 16.