

BLOCK MIGRATION

TECHNICAL FIELD

[0001] This disclosure relates to the field of blockchain technologies, and specifically, to a method and an apparatus for migrating a block.

BACKGROUND

[0002] Currently, a block migration technology based on a trusted execution environment (TEE) server is provided. If a stored block needs to be migrated from a blockchain system to another blockchain system, the TEE server pulls a to-be-migrated block from the blockchain system, and then verifies authenticity of the obtained to-be-migrated block. The authenticity of the block is used to indicate whether the block is a block that is really stored in the blockchain system. The TEE server can store the to-be-migrated block in the blockchain system after determining that authenticity verification of the to-be-migrated block succeeds. Certainly, in some cases, if all blocks in the blockchain system need to be migrated to the another blockchain system, the blocks in the blockchain system can be migrated and stored in the another blockchain system one by one in the above-mentioned method. Correspondingly, a user can query, by using the another blockchain system, a block migrated from the blockchain system.

[0003] In the above-mentioned migration solution, the authenticity of the block can be verified based on verification logic embedded in the TEE server. However, such a block authenticity verification manner is invisible to the user, and the user cannot learn of content included in the verification logic of verifying the authenticity of the block in the TEE server and specific aspects in which the authenticity of the block is verified. Further, the user cannot determine whether the TEE server verifies the authenticity of the block. Therefore, the user is unable, to some extent, to trust the authenticity of the blocks migrated in the above-mentioned migration method.

SUMMARY

[0004] In view of this, embodiments of this disclosure are intended to provide a method and an apparatus for migrating a block, to improve transparency of a method for verifying authenticity of the block in a block migration process.

- [0005]** According to a first aspect, a method for migrating a block is provided. The block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the method is applied to a blockchain node in the target blockchain, the target blockchain stores a first smart contract, and the first smart contract is used to manage a block in the source blockchain. The method includes: receiving a first invocation request for the first smart contract, where the first invocation request includes the first block in the source blockchain; verifying authenticity of the first block based on the first smart contract; and storing the first block in the target blockchain if verification performed on the authenticity of the first block succeeds.
- [0006]** According to a second aspect, a method for migrating a block is provided. The block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the target blockchain stores a first smart contract, the first smart contract is used to manage a block that is in the source blockchain and that is forwarded by a forwarding device, and the method is applied to the forwarding device. The method includes: obtaining the first block from the source blockchain; and sending a first invocation request for the first smart contract to the target blockchain node, where the first invocation request includes the first block.
- [0007]** According to a third aspect, an apparatus for migrating a block is provided. The block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the apparatus is applied to a blockchain node in the target blockchain, the target blockchain stores a first smart contract, and the first smart contract is used to manage a block in the source blockchain. The apparatus includes: a receiving module, configured to receive a first invocation request for the first smart contract, where the first invocation request includes the first block in the source blockchain; and a processing module, configured to verify authenticity of the first block based on the first smart contract. The processing module is further configured to store the first block in the target blockchain if authenticity verification of the first block succeeds.
- [0008]** According to a fourth aspect, an apparatus for migrating a block is provided. The block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the target blockchain stores a first smart contract, the first smart contract is used to manage a block that is in the source blockchain and that is forwarded by a forwarding device, and the apparatus is applied to the forwarding device. The apparatus includes: an obtaining module, configured to obtain the first block from the source blockchain; and a sending module, configured to send a first invocation request for the first smart contract to the target blockchain node, where the first invocation request includes the first block.
- [0009]** According to a fifth aspect, an apparatus for migrating a block is provided, including a memory and a processor. The memory stores executable code, and the processor is configured to

execute the executable code, to implement the method in the first aspect or the second aspect.

[0010] According to a sixth aspect, a computer-readable storage medium is provided. The computer-readable storage medium stores executable code, and when the executable code is executed, the method in the first aspect or the second aspect can be implemented.

5 [0011] According to a seventh aspect, a computer program product is provided, including executable code, and when the executable code is executed, the method in the first aspect or the second aspect can be implemented.

[0012] In this disclosure, the first smart contract is deployed in the target blockchain to verify the authenticity of the block. In this way, a user can view a block authenticity verification method
10 by querying the first smart contract deployed in the target blockchain, to help improve transparency of the verification method and improve a degree of trust of the user in a migration process.

BRIEF DESCRIPTION OF DRAWINGS

[0013] FIG. 1 is a schematic diagram illustrating a blockchain system to which an embodiment of this disclosure is applicable;

15 [0014] FIG. 2 is a schematic structural diagram illustrating a block to which an embodiment of this disclosure is applicable;

[0015] FIG. 3 is a schematic diagram illustrating a conventional block migration scenario;

[0016] FIG. 4 is a schematic diagram illustrating a migration scenario to which an embodiment of this disclosure is applicable;

20 [0017] FIG. 5 is a flowchart illustrating a method for migrating a block, according to an embodiment of this disclosure;

[0018] FIG. 6 is a schematic diagram illustrating a first smart contract, according to an embodiment of this disclosure;

[0019] FIG. 7 is a schematic diagram illustrating a block migration scenario, according to
25 another embodiment of this disclosure;

[0020] FIG. 8 is a flowchart illustrating a method for migrating a block, according to another embodiment of this disclosure;

[0021] FIG. 9 is a schematic diagram illustrating an apparatus for migrating a block, according to an embodiment of this disclosure;

30 [0022] FIG. 10 is a schematic diagram illustrating an apparatus for migrating a block, according to an embodiment of this disclosure; and

[0023] FIG. 11 is a schematic block diagram illustrating an apparatus, according to another embodiment of this disclosure.

DESCRIPTION OF EMBODIMENTS

[0024] The following clearly and completely describes the technical solutions in the embodiments of this disclosure with reference to the accompanying drawings in the embodiments of this disclosure. Apparently, the described embodiments are merely some rather than all of the
5 embodiments of this disclosure. To facilitate understanding of the solutions of this disclosure, the following first describes, with reference to FIG. 1, a blockchain system to which this disclosure is applicable and some terms used in this disclosure.

[0025] Blockchain system:

[0026] FIG. 1 is a schematic diagram illustrating a blockchain system to which an embodiment
10 of this disclosure is applicable. A blockchain system 100 shown in FIG. 1 includes a blockchain node 110 and a user node 120.

[0027] The blockchain (Blockchain) system is a typical distributed coordination system. A plurality of parties jointly maintain continuously increasing distributed data records. Content and a time sequence of data of the records are protected by using a cryptography technology, so that it is
15 difficult for any party to tamper with, deny, or forge the content and the time sequence.

[0028] In the blockchain system, data (for example, transaction information and a transaction execution result) can be encapsulated in a form of a block (Block). Blocks can be chained to each other by using a forward application, to form a "chain", that is, a blockchain. Usually, a block before a current block in the blockchain is referred to as a "previous block" and a block after the
20 current block in the blockchain is referred to as a "successor block". The following describes the block with reference to FIG. 2. For brevity, details are omitted here.

[0029] Usually, the user node 120 in the blockchain can communicate with the blockchain node 110 through a client. In some cases, the user node 120 can send, to one or more associated blockchain nodes 110 by submitting a transaction (Transaction), data that needs to be chained. The
25 transaction can be understood as a transaction request in the blockchain, and is used to carry a structure of specific service operation data.

[0030] It should be noted that all change operations for a world state in the blockchain can be performed based on the transaction. The world state can be understood as a storage state of a blockchain account, and includes a basic storage state of all accounts and an internal storage state of
30 a contract account. A contract platform can be understood as a "state machine" of the transaction. The world state describes a current basic storage state. A smart contract is executed, so that the world state may change to another new world state.

[0031] Correspondingly, after verifying authenticity of the data, the blockchain node 110 can broadcast the data to each blockchain node 110 in a network-wide broadcast manner. In this way,

each blockchain node 110 records full and consistent data.

[0032] In some implementations, the blockchain node 110 can be a device with a computing capability, for example, a server, a server group, or a blockchain chip. The server group can be centralized or can be distributed. In another implementation, the server can alternatively be a server that provides a service for a cloud platform.

[0033] In some implementations, the user node 120 can be a device with a computing capability, for example, can be a server, a terminal, or a mobile terminal.

[0034] Block:

[0035] As described above, a blockchain is a chained storage structure, and the block is a data element in the chained storage structure. The blockchain can be understood as a unidirectional chained structure in which blocks are connected to each other. Usually, the first block in the blockchain is referred to as a "genesis block" or an "initial block".

[0036] Usually, each block can record basic data such as a hash value of a previous block, transaction information in the current block, and hash of the current block. FIG. 2 shows a structure of a block to which an embodiment of this disclosure is applicable. As shown in FIG. 2, the block 200 can include two parts: a block header 210 and a block body 220.

[0037] The block header 210 can include basic information of the current block, to ensure that the current block can correctly enter the blockchain. In some implementations, the block header 210 can record a block hash (Hash) value of a previous block of the current block. Currently, the block hash value of the previous block can be recorded in a "prevBlockHash" field in the block header 210. In some other implementations, the block header 210 can further record a block height of the current block. The block height is used to identify a location of a block in the blockchain. Usually, a block height of the genesis block is 0.

[0038] The block body 220 can be used to record the transaction information, including a transaction quantity and transaction data.

[0039] Consensus mechanism:

[0040] The consensus mechanism can be understood as a manner of reaching a consensus among all accounting nodes to determine validity of a record. This is both a determining means and a tamper-proofing means. Four different consensus mechanisms is provided for the blockchain, and are applicable to different application scenarios, to achieve a balance between efficiency and security.

[0041] The consensus mechanism of the blockchain is featured by "minority is subordinate to majority" and "everyone is equal". Here, "minority is subordinate to majority" does not completely refer to a quantity of nodes, can be a calculation capability, a quantity of shareholdings, or another

computer comparable feature quantity. "Everyone is equal" means that when a node satisfies a condition, all nodes have the right to preferentially provide a consensus result, and the consensus result is directly agreed on by another node and finally becomes a final consensus result. A bitcoin is used as an example. A proof of work is used. A non-existent record may be forged only when at least 51% of accounting nodes in an entire network are controlled. When enough nodes join the blockchain, it is basically impossible to forge a non-existent record, to eliminate forging possibility.

[0042] Self-trust of the blockchain is mainly reflected in that a user distributed in the blockchain does not need to trust the other party of a transaction, and does not need to trust a centralized organization. The transaction can be implemented provided that a software system in a blockchain protocol is trusted. Such self-trust is achieved when a consensus (consensus) mechanism of the blockchain exists. To be specific, in a market of mutual distrust, a sufficient and necessary condition for all nodes to reach an agreement is that each node follows a rule preset in a protocol spontaneously and honestly for maximum benefit of the node, determines authenticity of each record, and finally records a determined real record in the blockchain. In other words, if all nodes have respective separate interests and contend with each other, it is almost impossible for the nodes to conspire to make a fraud, and it is particularly evident when the nodes have public reputation in the network. A blockchain technology is to use a set of consensus-based mathematical algorithms to build a "trust" network between machines, to give an endorsement by using a technology instead of creating a new credit by using a centralized credit institution.

[0043] A current consensus mechanism of the blockchain can include a proof of work (PoW) mechanism, a proof of stake mechanism, a deposit-based proof of stake mechanism, a verification pool mechanism, and a practical byzantine fault tolerance (PBFT).

[0044] Smart contract (Smart Contracts):

[0045] The smart contract is a set of commitments defined in a digital form, and includes a protocol in which a contract participant can execute the commitments. Alternatively, the smart contract can be understood as a program deployed on a computer system. When a trigger condition of the smart contract is satisfied, the smart contract can be automatically executed.

[0046] A blockchain provides technical support for an implementation of the smart contract. The smart contract is written into the blockchain in the digital form, and features of a blockchain technology ensure that an entire process of storing, reading, and executing the smart contract is transparent and traceable, and cannot be tampered with. In addition, a state machine system can be constructed by using a consensus algorithm of the blockchain, so that the smart contract can run efficiently.

[0047] In some implementations, a user can invoke the smart contract by submitting a

transaction to a blockchain system, set data recorded in the smart contract, and store the specified smart contract in the blockchain. Correspondingly, when a specific condition of the smart contract is triggered, a blockchain node can execute the smart contract, and record an execution result of the smart contract and an execution state of the smart contract.

5 [0048] Currently, various industries and even certain fields (for example, finance, public interest, insurance, and cross-border payment) in the industry build different blockchain systems based on respective industrial structures, and record valuable information and assets in the industry in the blockchain system.

10 [0049] However, in some scenarios, information and assets of no industry can be completely isolated, and there is usually a service requirement for cross-industry coordination. Therefore, the information and the assets recorded in the blockchain system need to be migrated between a plurality of blockchain systems, or a block stored in the blockchain system needs to be migrated between a plurality of blockchain systems. This means that a "trusted connection technology" is required, to truly and reliably migrate a block in a blockchain network to another blockchain
15 network, and ensure that content of the block cannot be tampered with in a block migration process. In this way, a participant in the another blockchain network can trust the migrated block, and perform service coordination by using the migrated block.

[0050] In some other scenarios, the blockchain system needs to be upgraded. However, for some reasons (for example, the upgraded blockchain system no longer supports a block format
20 supported by the blockchain system existing before upgrade), upgrade cannot be performed when a block is stored, in other words, online upgrade cannot be performed. Therefore, the block stored in the blockchain system needs to be migrated to the another blockchain system, to upgrade the blockchain system. This also means that a "trusted connection technology" is required, to truly and reliably migrate a block in a to-be-upgraded blockchain network to another blockchain network, and
25 ensure that content of the block cannot be tampered with in a block migration process. In this way, the to-be-upgraded blockchain network can be upgraded, and the user can continue to use the migrated block by using the another blockchain system.

[0051] A block migration technology based on a trusted execution environment (TEE) server is provided currently, to meet a requirement for cross-chain migration of a block in each scenario. The
30 following provides descriptions with reference to FIG. 3.

[0052] In a scenario shown in FIG. 3, it is assumed that a stored block needs to be migrated from a blockchain system 310 to a blockchain system 320. A TEE server 330 pulls a to-be-migrated block from the blockchain system 310, and then verifies authenticity of the obtained to-be-migrated block. The authenticity of the block is used to indicate whether the block is a block that is really

stored in the blockchain system 310. The TEE server can store the to-be-migrated block in the blockchain system 320 after determining that authenticity verification of the to-be-migrated block succeeds. Certainly, in some cases, if all blocks in the blockchain system 310 need to be migrated to the blockchain system 320, the blocks in the blockchain system 310 can be migrated and stored in the blockchain system 320 one by one in the above-mentioned method. Correspondingly, a user can query, by using the blockchain system 320, a block migrated from the blockchain system 310.

[0053] In the above-mentioned migration solution, the authenticity of the block can be verified based on verification logic embedded in the TEE server. However, such a block authenticity verification manner is invisible to the user, and the user cannot learn of content included in the verification logic of verifying the authenticity of the block in the TEE server and specific aspects in which the authenticity of the block is verified. Further, the user cannot determine whether the TEE server verifies the authenticity of the block. Therefore, the user is unable, to some extent, to trust the authenticity of the blocks migrated in the above-mentioned migration method.

[0054] To avoid the above-mentioned problem, and improve transparency of a method for verifying authenticity of a block in a block migration process, this disclosure provides a method for migrating a block. In a scenario in which the block is migrated from a source blockchain to a target blockchain, a first smart contract is deployed in the target blockchain to verify the authenticity of the block. In this way, a user can view a block authenticity verification method by querying the first smart contract deployed in the target blockchain, to help improve transparency of the verification method and improve a degree of trust of the user in a migration process. It should be noted that, for ease of description, a current block in the migration process is referred to as a "first block" below.

[0055] Before a method in the embodiments of this disclosure is described with reference to FIG. 5, a migration scenario to which the embodiments of this disclosure are applicable is described below with reference to FIG. 4. It should be noted that a source blockchain 410 and a target blockchain 420 shown in FIG. 4 can be blockchains in the blockchain system shown in FIG. 1. For related content, references can be made to the above-mentioned descriptions. For brevity, details are omitted here.

[0056] As shown in FIG. 4, it is assumed that a first block needs to be migrated from the source blockchain 410 to the target blockchain 420, and a first smart contract 421 is deployed in the target blockchain 420. The first smart contract 421 is configured to manage a block in the source blockchain 410.

[0057] FIG. 5 is a flowchart illustrating a method for migrating a block, according to an embodiment of this disclosure. The method shown in FIG. 5 includes step S510 to step S530. The steps shown in FIG. 5 can be performed by a blockchain node in the target blockchain 420.

[0058] In step S510, a first invocation request for the first smart contract 421 is received.

[0059] In some implementations, the first invocation request is used to request to migrate the first block from the source blockchain 410 to the target blockchain 420. The first invocation request includes the first block in the source blockchain 410.

5 [0060] In step S520, authenticity of the first block is verified based on the first smart contract 421.

[0061] The authenticity of the first block is used to indicate whether the first block is a block stored in the source blockchain 410. In some implementations, the authenticity of the first block can be verified based on validity of the first block. In some other implementations, the authenticity of

10 the first block can be verified based on validity of a blockchain node of the first block. In some other implementations, the authenticity of the first block can be verified based on data content in the first block. The following describes in detail a method for verifying the authenticity of the first block. For brevity, details are omitted here.

[0062] In step S530, the first block is stored in the target blockchain 420 if authenticity

15 verification of the first block succeeds.

[0063] If authenticity verification of the first block succeeds, it can be considered to a certain extent that the first block is a block stored in the source blockchain 410. In this case, the first block can be stored in the target blockchain 420. Alternatively, the first block is persisted into the target blockchain 420.

20 [0064] In some cases, authenticity verification of the first block may fail, that is, the first block may not be a block stored in the source blockchain 410. In this case, an invocation process of the first smart contract 421 can be ended. In other words, current invocation of the first smart contract 421 is ended. In other words, the method further includes: step S540: If authenticity verification of the first block fails, the invocation process of the first smart contract 421 is ended.

25 [0065] As described above, the authenticity of the first block needs to be verified, to ensure that the first block is a block in the source blockchain 410. The following separately describes four manners of verifying the authenticity of the first block in this embodiment of this disclosure.

[0066] Verification manner 1: The authenticity of the first block can be verified based on whether the first block is a successor block of a successfully migrated block. For ease of description,

30 the successfully migrated block is referred to as a "second block" below.

[0067] To be specific, the target blockchain 420 stores the second block migrated from the source blockchain 410, and the second block is a previous block of the first block in the source blockchain 410. In this case, step S520 includes: determining, based on the first smart contract 421, whether the first block is a successor block of the second block, to verify the authenticity of the first

block.

[0068] In the above-mentioned descriptions of a block, if the first block is a successor block of the second block, a block header of the first block records a block hash value of the second block. Therefore, in some implementations, a block hash value of the second block can be recorded based on the first smart contract 421, and the block hash value of the second block is compared with the block hash value that is of the second block and that is recorded in the first block. If the block hash value that is of the second block and that is recorded in the first smart contract 421 is the same as the block hash value that is of the second block and that is recorded in the first block, it can be considered that the first block is a successor block of the second block, that is, authenticity verification of the first block succeeds. On the contrary, if the block hash value that is of the second block and that is recorded in the first smart contract 421 is different from the block hash value that is of the second block and that is recorded in the first block, it can be considered that the first block is not a successor block of the second block, that is, authenticity verification of the first block fails.

[0069] It should be noted that the block hash value that is of the second block and that is recorded in the first smart contract 421 can be recorded in a process of migrating the second block. For example, the block hash value that is of the second block and that is recorded in the first smart contract 421 can be obtained by the following mentioned process in which a forwarding device migrates the second block. To be specific, after pulling the second block from the source blockchain, the forwarding device reads the block hash value in the second block and sends the block hash value to the first smart contract 421 in the target blockchain.

[0070] In a process of verifying the authenticity of the first block in the verification manner 1, if the first block is the first block in the source blockchain 410, the verification manner 1 may be unable to be used. To avoid this problem, information about a genesis block in the source blockchain 410 can be recorded in the first smart contract 421, so that for another block in the source blockchain 410, authenticity of the block can be verified in the verification manner 1. In some implementations, the information about the genesis block in the source blockchain 410 can be entered in an initialization process of the first smart contract 421.

[0071] The information about the genesis block can include a block hash value of the genesis block. Certainly, in some cases, the information about the genesis block can further include block content of the genesis block, to verify whether the block hash value that is of the genesis block and that is recorded in the first smart contract 421 is obtained based on the block content of the genesis block, thereby improving authenticity of the information about the genesis block.

[0072] Verification manner 2: The authenticity of the first block can be verified based on a maximum value that is of the block hash value and that is agreed on by blockchain nodes.

[0073] Usually, all blockchain nodes in a blockchain network agree on a maximum value of the block hash value, to specify that a block hash value of a block in the blockchain system cannot exceed the maximum value of the block hash value. Based on this, in this embodiment of this disclosure, the authenticity of the first block can be verified based on the maximum value of the block hash value.

[0074] To be specific, the first smart contract 421 records a maximum value that is of the block hash value and that is agreed on by blockchain nodes in the source blockchain 410. Step S520 can include: determining, based on the first smart contract 421, whether the block hash value of the first block is less than the maximum value of the block hash value, to verify the authenticity of the first block.

[0075] In some implementations, the block hash value of the second block can be compared with the maximum value of the block hash value based on the first smart contract 421. If the block hash value of the second block is less than the maximum value of the block hash value, it can be considered that the block hash value of the first block satisfies an agreement of the blockchain nodes in the source blockchain 410, and authenticity verification of the first block succeeds. On the contrary, if the block hash value of the second block is greater than the maximum value of the block hash value, it can be considered that the block hash value of the first block does not satisfy an agreement of the blockchain nodes in the source blockchain 410, that is, authenticity verification of the first block fails.

[0076] It should be noted that the maximum value that is of the block hash value and that is recorded in the first smart contract 421 can be entered in the initialization process of the first smart contract 421.

[0077] Verification manner 3: The authenticity of the first block can be verified based on the blockchain node of the first block in the source blockchain.

[0078] To be specific, the blockchain node that stores the first block in the source blockchain 410 is a first blockchain node, the first block is a block to which a digital signature is added by the blockchain node of the first block, and a public key of the first blockchain node is recorded in the first smart contract 421. Step S520 can include: determining, based on the first smart contract 421 and the public key of the first blockchain node, whether the digital signature of the first block is a digital signature of the first blockchain node, to verify the authenticity of the first block.

[0079] In some implementations, when the first smart contract 421 is invoked, the first smart contract 421 can be notified of a blockchain node of the to-be-migrated first block in the source blockchain, that is, the above-mentioned first blockchain node. In this way, the first smart contract 421 can verify the authenticity of the first block based on the recorded public key of the first

blockchain node and the digital signature on the first block. If the public key that is of the first blockchain node and that is recorded in the first smart contract 421 matches the digital signature on the first block, it can be considered that the first block is a block transmitted by the first blockchain node, and authenticity verification of the first block succeeds. On the contrary, if the public key that is of the first blockchain node and that is recorded in the first smart contract 421 does not match the digital signature on the first block, it can be considered that the first block is not a block transmitted by the first blockchain node, and authenticity verification of the first block fails.

[0080] It should be noted that the public key of the first blockchain node in the source blockchain can be entered in the initialization process of the first smart contract 421.

10 [0081] In some other implementations, the first smart contract 421 can further record public keys of some or all blockchain nodes in the source blockchain, to verify the authenticity of the block in a process of migrating blocks of different blockchain nodes. Certainly, the public keys of the some or all of blockchain nodes can also be entered in the initialization process of the first smart contract 421.

15 [0082] Verification manner 4: The authenticity of the first block is verified based on transaction information of a block stored in another blockchain node in the source blockchain.

[0083] In an actual case, a blockchain may fork in the source blockchain 410. In other words, a plurality of blocks in which different transaction information is recorded contend for a location of a same block height. However, only one block obtains the location of the block height through contention, and a consensus is reached by the blockchain nodes in the entire source blockchain 410. For example, a block 1 and a block 2 in the source blockchain 410 simultaneously contend for a location of a block height 2, the block 1 records content of the transaction 1, the transaction 3, and the transaction 5, and the block 2 records content of the transaction 2, the transaction 4, and the transaction 6. Finally, only the block 1 obtains a location of the block 2 through contention, and a consensus is reached by all the blockchain nodes in the source blockchain 410.

20 [0084] In the above-mentioned scenario, although all the blockchain nodes in the source blockchain 410 reach no consensus on the block 2, a block hash value of the block 2 is less than the agreed maximum value of the block hash value in the source blockchain 410, and a block hash value of a block at a location of a block height 1 is recorded in the block 2. In this case, if a blockchain node of the block 2 does evil, a digital signature is added to the block 2 and the block 2 is migrated to the target blockchain node 420. Authenticity of the block 2 is verified in any one of the verification manner 1 to the verification manner 3, and an obtained verification result is that verification success. In this case, that the block 2 is actually not a block on which a consensus is reached in the source blockchain 410 cannot be verified in the verification manner.

[0085] Therefore, to improve block authenticity verification accuracy, this embodiment of this disclosure further provides a verification manner 4 of block authenticity. To be specific, the first smart contract 421 records first transaction information of a third block stored in the second blockchain node in the source blockchain 410, and a block height of the third block in the source blockchain 410 is the same as a block height of the first block. Step S520 further includes: comparing transaction information recorded in the first block and the first transaction information based on the first smart contract 421, to verify the authenticity of the first block.

[0086] If content of the first transaction information is the same as content of the transaction information recorded in the first block, it can be understood that the first block is a block on which a consensus is reached in the source blockchain 410, and authenticity verification of the first block succeeds. On the contrary, if content of the first transaction information is different from content of the transaction information recorded in the first block, it can be understood that the first block is a block on which no consensus is reached in the source blockchain 410, and authenticity verification of the first block fails.

[0087] It should be noted that the first transaction information and/or the transaction information recorded in the first block can be obtained in a migration process of a corresponding block. For example, the first transaction information can be obtained in the following process in which the forwarding device migrates the third block. To be specific, after pulling the third block from the source blockchain, the forwarding device reads the first transaction information in the third block and sends the first transaction information to the first smart contract 421 in the target blockchain. Correspondingly, the transaction information recorded in the first block can also be obtained by the following process in which the forwarding device migrates the first block. To be specific, after pulling the first block from the source blockchain, the forwarding device reads the transaction information in the first block and sends the transaction information to the first smart contract 421 in the target blockchain.

[0088] Usually, transaction information recorded in a block that is at a corresponding location and that is stored in each blockchain node in the source blockchain can be compared based on the first smart contract 421, to further improve accuracy of verifying the authenticity of the block. Only when transaction information in a block that is at a corresponding location and that is stored in a preset quantity of blockchain nodes is the same, it is determined that the block in which the transaction information is recorded is a block on which a consensus is reached in the source blockchain 410, and authenticity of the block is verified. Blocks at corresponding locations can be understood as blocks with a same block height.

[0089] For example, when it is specified in the first smart contract 421 that transaction

information recorded in a block that is at a corresponding location and that is stored in one third of the blockchain nodes in the source blockchain 410 is the same, it can be considered that a consensus is reached on the block in which the transaction information is recorded. It is assumed that there are a total of 3000 blockchain nodes in the source blockchain 410, and a block height of a block 3 is 4.

5 In this case, based on a specification in the first smart contract 421, if transaction information recorded in the block 3 whose block height is 4 in 1000 blockchain nodes in the source blockchain 410 is completely the same, it can be considered that the block 3 in which the transaction information is recorded is a real block in the source blockchain 410, that is, block authenticity verification succeeds.

10 **[0090]** It should be noted that, if a plurality of blocks in which different transaction information is recorded exist in a comparison process of blockchain nodes whose quantity does not reach a preset quantity, and block heights of the blocks are the same, content recorded in the plurality of blocks can be retained, and quantities corresponding to blocks with different transaction information are separately recorded in a subsequent process of verifying a corresponding block in
15 another blockchain node. Until a quantity of blockchain nodes corresponding to a block in which certain transaction information is recorded reaches the preset quantity, it is considered that authenticity verification of the block with the transaction information succeeds.

[0091] For example, when authenticity verification is performed on a block 2 in the 1st blockchain node in the source blockchain 410 in the verification manner 4, transaction information recorded in the block 2 is transaction information 1. When authenticity verification is performed on
20 a block 2 in the 2nd blockchain node in the source blockchain 410 in the verification manner 4, the transaction information recorded in the block 2 is transaction information 2. In this case, because transaction information recorded in a block 2 stored in another blockchain node in the source blockchain is not learned of, the transaction information 1 and the transaction information 2 can be
25 separately retained; and in a subsequent process of verifying authenticity of the block 2 in the another blockchain node in the source blockchain 410, a quantity of blockchain nodes corresponding to each piece of transaction information is recorded, and after the quantity of blockchain nodes reaches the preset quantity, the block 2 of the transaction information is used as a real block recorded in the source blockchain 410, that is, authenticity verification of the block
30 succeeds.

[0092] It should be noted that a total quantity of nodes in the source blockchain and/or the preset quantity can be set in an initialization process of the first smart contract 421. Certainly, the above-mentioned parameters can alternatively be specified in a process of invoking the first smart contract 421. This is not limited in this embodiment of this disclosure.

[0093] In addition, the preset quantity can be specified based on a consensus mechanism in the source blockchain. For example, when the consensus mechanism used in the source blockchain 410 is PBFT, the preset quantity can be one third of a total quantity of blockchain nodes in the source blockchain 410. For another example, when a used consensus mechanism is a PoW, the preset
5 quantity can be one half of a total quantity of blockchain nodes in the source blockchain 410. Certainly, in this embodiment of this disclosure, the preset quantity can alternatively be set based on experience. This is not limited in this embodiment of this disclosure.

[0094] The verification manner 1 to the verification manner 4 are separately described in the above-mentioned descriptions. It should be noted that, in this embodiment of this disclosure, the
10 four verification manners can be used in combination with each other. For example, the verification manner 1 and the verification manner 2 can be used in combination. For another example, the verification manner 1, the verification manner 3, and the verification manner 4 can also be used in combination. For brevity, details are not listed one by one in this embodiment of this disclosure. In
15 addition, for ease of understanding, the following describes, with reference to FIG. 8, a method procedure in which the four verification manners are used in combination.

[0095] In addition, it should be further noted that, when the four verification manners can be used in combination, a sequence of the verification manners is not limited in this embodiment of this disclosure. For example, a verification manner 1 can precede a verification manner 2. For another example, a verification manner 3 can precede a verification manner 2.

[0096] To help a user to query a block in the source blockchain 410, a query interface can be set in the first smart contract 421, so that a user of the target blockchain 420 invokes the query interface. That is, the method further includes: receiving a second invocation request for the first smart contract 421 from a user node of the target blockchain 420. The second invocation request is used to query a block that is in the source blockchain 410 and that is stored in the target blockchain
25 420.

[0097] It should be noted that the block that is in the source blockchain 410 and that is stored in the target blockchain 420 can be a block whose block authenticity verification succeeds. In some implementations, to prevent a blockchain fork from existing in the source blockchain 410, a block (that is, no consensus is reached on the block) in a shorter chain obtained after the source
30 blockchain forks is migrated to the target blockchain, so that the user finds the block in the shorter chain in the target blockchain. It can be set in the first smart contract 421 that after an authenticity verification process of a block whose block height is N succeeds, the above-mentioned query service can be provided for a block whose block height is $(N-i)$ in the source blockchain, to ensure query reliability. N and i are positive integers, and N is greater than i . In some cases, i can be set to

6 based on experience. That is, after authenticity verification of a block whose block height is i succeeds, a query service of the block whose block height is $N-i$ is provided, to help reduce a possibility that the user finds the block in the shorter chain.

[0098] For ease of understanding, the following describes content of the first smart contract in this embodiment of this disclosure with reference to FIG. 6. FIG. 6 is a schematic diagram illustrating a first smart contract, according to an embodiment of this disclosure. As shown in FIG. 6, the first smart contract can include a transfer block (represented by "transferBlock") interface and a query block (represented by "queryBlock") interface.

[0099] The transfer block interface can be configured to upload the block in the source blockchain 410. That is, the first invocation request can be understood as invoking the transfer block interface. When the transfer block interface is invoked, in addition to a to-be-uploaded block (represented by "Block"), a blockchain node (represented by "node") that stores the block needs to be specified.

[0100] The query block interface can be configured to query a block whose authenticity verification succeeds. That is, the second invocation request can be understood as invoking the query block interface. When the query block interface is invoked, an index of a to-be-queried block (represented by "blockNumber") needs to be specified.

[0101] Persistent data of the first smart contract includes a block list (represented by "List<Block>") and blockchain node information in the source blockchain (represented by "Map<node, NodeInfo>"). The list records a block whose authenticity verification succeeds. The blockchain node information can include an identifier (represented by "node") of a blockchain node and a public key (represented by "NodeInfo") of the blockchain node.

[0102] In some implementations, the source blockchain 410 and the target blockchain 420 cannot directly communicate with each other to migrate the first block. In this case, a forwarder can be set between the source blockchain 410 and the target blockchain 420 to migrate the first block from the source blockchain 410 to the target blockchain 420. That is, the forwarding device obtains a to-be-migrated block from the source blockchain 410, and then the forwarding device sends the first invocation request to a blockchain node in the target blockchain 420. Therefore, step S510 can include: receiving the first invocation request for the first smart contract 421 from the forwarding device. The forwarding device is configured to migrate the first block from the source blockchain 410 to the target blockchain 420.

[0103] In some implementations, the forwarding device can obtain the first block from the source blockchain 410 in a block pulling and parsing manner. In some other implementations, the forwarding device can obtain the first block by invoking a block query interface of a blockchain

node in the target blockchain 420. A manner in which the forwarding device obtains the first block is not limited in this embodiment of this disclosure.

[0104] In some other implementations, the forwarding device can be a device with a processing function, for example, can be a server, a terminal device, or a miner. It should be noted that, in this embodiment of this disclosure, the authenticity of the first block is verified based on the first smart contract, to ensure authenticity of a block recorded in the target blockchain. Compared with a conventional block migration solution, the solution in this embodiment of this disclosure does not strongly depend on whether the forwarding device is a TEE server. Certainly, the forwarding device can also be configured as a TEE server, to further improve security in a block migration process.

[0105] To avoid a block migration failure caused when the forwarding device does evil, a plurality of forwarding devices can be disposed between the source blockchain 410 and the target blockchain 420, to migrate the first block from the source blockchain 410 to the target blockchain 420. Certainly, if a problem such as reduction in costs is considered, a forwarding device can also be disposed between the source blockchain 410 and the target blockchain 420.

[0106] In some implementations, an incentive mechanism can be set, to encourage the forwarding device to migrate a block from the source blockchain 410 to the target blockchain 420. In some implementations, the forwarding device can be incentivized in an on-chain manner. For example, an incentive payment can be stored in an on-chain account (for example, an account of the first smart contract is created), and the incentive payment in the account is distributed to an on-chain account corresponding to the forwarding device based on a migration success rate and/or a quantity of successfully migrated blocks of each forwarding device. Factors referenced by the incentive mechanism are not specifically limited in this disclosure. For example, the incentive payment can alternatively be allocated based on a quantity of times that the forwarding device invokes the first smart contract.

[0107] In some other implementations, the forwarding device can be incentivized in an off-chain manner. For example, the migration success rate and/or the quantity of successfully migrated blocks of each forwarding device can be recorded on a chain, and then, based on these records, the incentive payment is allocated to an off-chain account corresponding to the forwarding device in an off-chain manner. Factors referenced by the incentive mechanism are not specifically limited in this disclosure. For example, the incentive payment can alternatively be allocated based on a quantity of times that the forwarding device invokes the first smart contract.

[0108] The method in the embodiments of this disclosure is described above with reference to FIG. 1 to FIG. 6. For ease of understanding, the following describes, with reference to FIG. 7 to FIG. 8 by using migration of a block 1 as an example, a procedure of the method for migrating a

block in the embodiments of this disclosure. As shown in FIG. 7, it is assumed that the block 1 is stored in a blockchain node 1 in a source blockchain 710, and a block height of the block 1 is 2. A forwarding device 730 is configured to migrate the block 1 from the source blockchain 710 to a target blockchain 720. Correspondingly, a smart contract 1 is deployed in the target blockchain 720, and the smart contract 1 is used to manage a block in the source blockchain 710.

[0109] FIG. 8 is a flowchart illustrating a method for migrating a block, according to another embodiment of this disclosure. The method shown in FIG. 8 includes step S810 and step S820.

[0110] In step S810, the smart contract 1 is initialized in the target blockchain 720.

[0111] The above-mentioned process of initializing the smart contract 1 includes recording, in the smart contract 1, information about a genesis block in the source blockchain 710, all blockchain node information of the source blockchain 710, and a maximum value of a block hash value agreed in the source blockchain 710. The information about the genesis block can include a block hash value of the genesis block and block content of the genesis block. The blockchain node information includes an identifier of a blockchain node and a public key of the blockchain node.

[0112] In step S820, the forwarding device 730 obtains a block 1 from a blockchain node 1 in the source blockchain.

[0113] In some implementations, the forwarding device 730 can find the block 1 by invoking a block query interface of the blockchain node 1. Correspondingly, full transaction information in the block 1 can be obtained. In addition, the block 1 is a block to which a digital signature is added by the blockchain node 1 based on a private key.

[0114] In step S830, the forwarding device 730 sends an invocation request 1 for the smart contract 1 to a blockchain node in the target blockchain 720.

[0115] In some implementations, the forwarding device 730 can invoke a transfer block interface of the smart contract 1 based on the invocation request 1, to send the block 1 to the target blockchain 720.

[0116] In step S840, the blockchain node in the target blockchain 720 verifies authenticity of the block 1 based on the smart contract 1.

[0117] In some implementations, the blockchain node verifies, in a verification manner 1 based on the smart contract 1, whether the block 1 is a successor block of a block 0. The block 0 is a previous block of the block 1 in the source blockchain 710. If the block 1 is a successor block of the block 0, the authenticity of the block 1 continues to be verified in a verification manner 2. If the block 1 is not a successor block of the block 0, current invocation of the smart contract 1 is ended.

[0118] If the block 1 is a successor block of the block 0, whether a block hash value of the block 1 is less than a maximum value that is of the block hash value and that is agreed in the source

blockchain 710 is verified in the verification manner 2. If the block hash value of the block 1 is less than the maximum value of the block hash value, the authenticity of the block 1 continues to be verified in a verification manner 3. If the block hash value of the block 1 is greater than or equal to the maximum value of the block hash value, current invocation of the smart contract 1 is ended.

5 [0119] If the block hash value of the block 1 is less than the maximum value of the block hash value, whether a blockchain node of the block 1 is the blockchain node 1 is verified in the verification manner 3. If the digital signature of the block 1 matches a public key of the blockchain node 1, the authenticity of the block 1 continues to be verified in the verification manner 4. If the digital signature of block 1 does not match a public key of the blockchain node 1, current invocation
10 of the smart contract 1 is ended.

[0120] If the digital signature of the block 1 matches the public key of the blockchain node 1, whether transaction information in the block 1 is the same as transaction information recorded in a block 1' is verified in the verification manner 4. The block 1' is a block that has a same block height as the block 1 and that is stored in a blockchain node 2 in the source blockchain 710, and a block
15 authenticity verification process of the block 1' succeeds. If the transaction information recorded in the block 1 is the same as the transaction information recorded in the block 1', authenticity verification of the block 1 succeeds, and step S850 is performed. If the transaction information recorded in the block 1 is different from the transaction information recorded in the block 1', current invocation of the smart contract 1 is ended.

20 [0121] In step S850, the block 1 is persisted in the smart contract 1 of the target blockchain 720.

[0122] In step S860, the blockchain node in the target blockchain 720 performs a query based on the smart contract 1, to determine whether respectively stored blocks 1 sent by a preset quantity of blockchain nodes in the source blockchain 710 are received.

25 [0123] If the respectively stored blocks 1 sent by the preset quantity of blockchain nodes in the source blockchain 710 are received, it can be understood that the block 1 is a block on which a consensus is reached in the source blockchain 710, and step S870 is performed. If the respectively stored blocks 1 sent by the preset quantity of blockchain nodes in the source blockchain 710 are not received, the forwarding device 730 is waited to continue to migrate a block 1 stored in another
30 blockchain node in the source blockchain 710, that is, step S880 is performed.

[0124] In step S870, a block height of the block on which a consensus is reached in the smart contract 1 is increased by 1, and the block 1 is added to a block list in the smart contract 1.

[0125] It should be noted that, a process of forwarding a certain block in the embodiments of this disclosure is described with reference to FIG. 4 and FIG. 5. If all blocks in the source

blockchain are to be migrated, the above-mentioned process can be cyclically performed until all the blocks in the source blockchain are migrated. For brevity, details are omitted here.

[0126] In addition, as described above, in a block authenticity verification process, full transaction information included in a certain block needs to be maintained, to facilitate execution of the verification manner 4. It should be noted that, to save storage space, when transaction information recorded in blocks is the same, only one piece of full data of the blocks can be recorded. Certainly, in this embodiment of this disclosure, transaction information that is of a block and that is stored in each blockchain node in the source blockchain can also be recorded.

[0127] The method embodiments of this disclosure are described in detail with reference to FIG. 1 to FIG. 8. The following describes apparatus embodiments of this disclosure in detail with reference to FIG. 9 to FIG. 11. It should be understood that descriptions of the method embodiments correspond to descriptions of the apparatus embodiments. Therefore, for a part that is not described in detail, references can be made to the method embodiments.

[0128] FIG. 9 is a schematic diagram illustrating an apparatus for migrating a block, according to an embodiment of this disclosure. The block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the apparatus is a blockchain node in the target blockchain, the target blockchain stores a first smart contract, the first smart contract is used to manage a block in the source blockchain, and the apparatus 900 includes a receiving module 910 and a processing module 920.

[0129] The receiving module 910 is configured to receive a first invocation request for the first smart contract. The first invocation request includes the first block in the source blockchain. The processing module 920 is configured to verify authenticity of the first block based on the first smart contract. The processing module 920 is further configured to store the first block in the target blockchain if authenticity verification of the first block succeeds.

[0130] In a possible implementation, the target blockchain stores a second block migrated from the source blockchain, and the second block is a previous block of the first block in the source blockchain; and the processing module is further configured to determine, based on the first smart contract, whether the first block is a successor block of the second block, to verify the authenticity of the first block.

[0131] In a possible implementation, the first smart contract records a maximum value that is of a block hash value and that is agreed on by a blockchain node in the source blockchain; and the processing module is further configured to determine, based on the first smart contract, whether a block hash value of the first block is less than the maximum value of the block hash value, to verify the authenticity of the first block.

[0132] In a possible implementation, a blockchain node that stores the first block in the source blockchain is a first blockchain node, the first block is a block to which a digital signature is added by the blockchain node of the first block, and the first smart contract records a public key of the first blockchain node; and the processing module is further configured to determine, based on the first smart contract and the public key of the first blockchain node, whether the digital signature of the first block is a digital signature of the first blockchain node, to verify the authenticity of the first block.

[0133] In a possible implementation, the first smart contract records first transaction information of a third block stored in a second blockchain node in the source blockchain, and a block height of the third block in the source blockchain is the same as a block height of the first block; and the processing module is further configured to compare transaction information recorded in the first block and the first transaction information based on the first smart contract, to verify the authenticity of the first block.

[0134] In a possible implementation, the receiving module is further configured to receive a second invocation request for the first smart contract from a user node of the target blockchain. The second invocation request is used to query a block that is in the source blockchain and that is stored in the target blockchain.

[0135] In a possible implementation, the receiving module is further configured to receive the first invocation request for the first smart contract from a forwarding device. The forwarding device is configured to migrate the first block from the source blockchain to the target blockchain.

[0136] In a possible implementation, the forwarding device is one of a plurality of forwarding devices, and the plurality of forwarding devices are configured to migrate the first block from the source blockchain to the target blockchain.

[0137] In a possible implementation, the processing module is further configured to end an invocation process of the first smart contract if authenticity verification of the first block fails.

[0138] FIG. 10 is a schematic diagram illustrating an apparatus for migrating a block, according to an embodiment of this disclosure. The block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the target blockchain stores a first smart contract, and the first smart contract is used to manage a block that is in the source blockchain and that is forwarded by a forwarding device. An apparatus 1000 in FIG. 10 is applied to the forwarding device, and includes an obtaining module 1010 and a sending module 1020.

[0139] The obtaining module 1010 is configured to obtain the first block from the source blockchain. The sending module 1020 is configured to send a first invocation request for the first smart contract to the target blockchain node. The first invocation request includes the first block.

[0140] FIG. 11 is a schematic block diagram illustrating an apparatus, according to another embodiment of this disclosure. An apparatus 1100 shown in FIG. 11 can include a memory 1110, a processor 1120, an input/output interface 1130, and a transceiver 1140. The memory 1110, the processor 1120, the input/output interface 1130, and the transceiver 1140 are connected through an internal connection path. The memory 1110 is configured to store instructions. The processor 1120 is configured to execute the instructions stored in the memory 1110, to control the input/output interface 1130 to receive input data and information, output data such as an operation result, and control the transceiver 1140 to send a signal.

[0141] The processor 1120 is configured to determine a dedicated transmission resource used to transmit a scheduling request SR. The transceiver 1140 is configured to send the SR to a network device on the dedicated transmission resource determined by the determining unit.

[0142] It should be understood that, in this embodiment of this disclosure, the processor 1120 can be a general-purpose central processing unit (CPU), a microprocessor, an application-specific integrated circuit (ASIC), or one or more integrated circuits, and is configured to execute a related program, to implement the technical solutions provided in the embodiments of this specification.

[0143] It should be further understood that, the transceiver 1140 is also referred to as a communication interface, and the transceiver 1140 implements communication between the apparatus 1100 and another device or a communication network by using a transceiver apparatus including but not limited to a transceiver.

[0144] The memory 1110 can include a read-only memory and a random access memory, and provides instructions and data for the processor 1120. A part of the processor 1120 can further include a nonvolatile random access memory. For example, the processor 1120 can further store information about a device type.

[0145] In an implementation process, steps in the methods can be implemented through an integrated logical circuit of hardware in the processor 1120, or by using instructions in a form of software. The method for migrating a block disclosed with reference to the embodiments of this disclosure can be directly performed and completed by a hardware processor, or can be performed and completed by a combination of hardware and a software module in the processor. The software module can be located in a mature storage medium in the art such as a random access memory, a flash memory, a read-only memory, a programmable read-only memory, an electrically erasable programmable memory, or a register. The storage medium is located in the memory 1110, and the processor 1120 reads information in the memory 1110 and completes the steps in the methods in combination with hardware of the processor 1120. To avoid repetition, details are omitted here.

[0146] It should be understood that, in this embodiment of this disclosure, the processor can be

a central processing unit (CPU), and the processor can be another general-purpose processor, a digital signal processor (DSP), an application-specific integrated circuit (ASIC), a field programmable gate array (FPGA) or another programmable logic device, a discrete gate or transistor logic device, a discrete hardware component, etc. The general-purpose processor may be a microprocessor, or the processor may be any conventional processor, etc.

[0147] It should be understood that in this embodiment of this disclosure, "B corresponding to A" indicates that B is associated with A, and B can be determined based on A. However, it should be further understood that determining B based on A does not mean that B is determined based on only A, and B may be further determined based on A and/or other information.

[0148] It should be understood that the term "and/or" in this specification describes only an association relationship for describing associated objects and represents that three relationships may exist. For example, A and/or B may represent the following three cases: Only A exists, both A and B exist, and only B exists. In addition, the character "/" in this specification usually indicates an "or" relationship between the associated objects.

[0149] It should be understood that sequence numbers of the above-mentioned processes do not mean execution sequences in various embodiments of this disclosure. The execution sequences of the processes should be determined based on functions and internal logic of the processes, and should not be construed as any limitation on the implementation processes of embodiments of this disclosure.

[0150] In the several embodiments provided in this disclosure, it should be understood that the disclosed system, apparatus, and method can be implemented in other manners. For example, the described apparatus embodiments are merely examples. For example, division into the units is merely logical function division and may be other division in an actual implementation. For example, a plurality of units or components may be combined or integrated into another system, or some features may be ignored or not performed. In addition, the displayed or discussed mutual couplings or direct couplings or communication connections may be implemented by using some interfaces. The indirect couplings or communication connections between the apparatuses or units may be implemented in electronic, mechanical, or other forms.

[0151] The units described as separate parts may or may not be physically separate, and parts displayed as units may or may not be physical units, that is, may be located at one place, or may be distributed on a plurality of network units. Some or all of the units may be selected based on an actual requirement, to achieve the objectives of the solutions of embodiments.

[0152] In addition, functional units in the embodiments of this disclosure can be integrated into one processing unit, each of the units can exist alone physically, or two or more units are integrated

into one unit.

[0153] All or some of the embodiments may be implemented by using software, hardware, firmware, or any combination thereof. When software is used to implement the embodiments, all or a part of the embodiments may be implemented in a form of a computer program product. The computer program product includes one or more computer program instructions. When the computer program instructions are loaded and executed on the computer, the procedure or functions according to embodiments of this disclosure are all or partially generated. The computer may be a general-purpose computer, a dedicated computer, a computer network, or other programmable apparatuses. The computer instructions may be stored in a computer-readable storage medium, or may be transmitted from a computer-readable storage medium to another computer-readable storage medium. For example, the computer instructions may be transmitted from a website, computer, server, or data center to another website, computer, server, or data center in a wired (for example, a coaxial cable, an optical fiber, or a digital subscriber line (DSL)) or wireless (for example, infrared, radio, or microwave) manner. The computer-readable storage medium can be any usable medium accessible by the computer, or a data storage device, for example, a server or a data center, integrating one or more usable media. The usable medium can be a magnetic medium (for example, a floppy disk, a hard disk, or a magnetic tape), an optical medium (for example, a digital video disc (DVD)), a semiconductor medium (for example, a solid state drive (SSD)), etc.

[0154] The above-mentioned descriptions are merely specific implementations of this disclosure, but the protection scope of this disclosure is not limited thereto. Any variation or replacement readily figured out by a person skilled in the art within the technical scope disclosed in this disclosure shall fall within the protection scope of this disclosure. Therefore, the protection scope of this disclosure shall be subject to the protection scope of the claims.

CLAIMS

What is claimed is:

1. A method for migrating a block, wherein the block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the method is applied to a blockchain node in the target blockchain, the target blockchain stores a first smart contract, and the first smart contract is used to manage a block in the source blockchain; and

the method comprises:

receiving a first invocation request for the first smart contract, wherein the first invocation request comprises the first block in the source blockchain;

verifying authenticity of the first block based on the first smart contract; and

storing the first block in the target blockchain if authenticity verification of the first block succeeds.

2. The method according to claim 1, wherein the target blockchain stores a second block migrated from the source blockchain, and the second block is a previous block of the first block in the source blockchain; and

the verifying authenticity of the first block based on the first smart contract comprises:

determining, based on the first smart contract, whether the first block is a successor block of the second block, to verify the authenticity of the first block.

3. The method according to claim 1 or 2, wherein the first smart contract records a maximum value that is of a block hash value and that is agreed on by a blockchain node in the source blockchain; and

the verifying authenticity of the first block based on the first smart contract comprises:

determining, based on the first smart contract, whether a block hash value of the first block is less than the maximum value of the block hash value, to verify the authenticity of the first block.

4. The method according to any one of claims 1 to 3, wherein a blockchain node that stores the first block in the source blockchain is a first blockchain node, the first block is a block to which a digital signature is added by the blockchain node of the first block, and the first smart contract records a public key of the first blockchain node; and

the verifying authenticity of the first block based on the first smart contract comprises:

determining, based on the first smart contract and the public key of the first blockchain node, whether the digital signature of the first block is a digital signature of the first blockchain node, to verify the authenticity of the first block.

5. The method according to any one of claims 1 to 4, wherein the first smart contract records first transaction information of a third block stored in a second blockchain node in the source

blockchain, and a block height of the third block in the source blockchain is the same as a block height of the first block; and

the verifying authenticity of the first block based on the first smart contract comprises:

5 comparing transaction information recorded in the first block and the first transaction information based on the first smart contract, to verify the authenticity of the first block.

6. The method according to any one of claims 1 to 5, wherein the method further comprises:

receiving a second invocation request for the first smart contract from a user node of the target blockchain, wherein the second invocation request is used to query a block that is in the source blockchain and that is stored in the target blockchain.

10 7. The method according to any one of claims 1 to 6, wherein the receiving a first invocation request for the first smart contract comprises:

receiving the first invocation request for the first smart contract from a forwarding device, wherein the forwarding device is configured to migrate the first block from the source blockchain to the target blockchain.

15 8. The method according to claim 7, wherein the forwarding device is one of a plurality of forwarding devices, and the plurality of forwarding devices are configured to migrate the first block from the source blockchain to the target blockchain.

9. The method according to any one of claims 1 to 8, wherein the method further comprises:

20 ending an invocation process of the first smart contract if authenticity verification of the first block fails.

10. A method for migrating a block, wherein the block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the target blockchain stores a first smart contract, the first smart contract is used to manage a block that is in the source blockchain and that is forwarded by a forwarding device, and the method is applied to the forwarding device; and

25 the method comprises:

obtaining the first block from the source blockchain; and

sending a first invocation request for the first smart contract to the target blockchain node, wherein the first invocation request comprises the first block.

30 11. An apparatus for migrating a block, wherein the block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the apparatus is applied to a blockchain node in the target blockchain, the target blockchain stores a first smart contract, and the first smart contract is used to manage a block in the source blockchain; and

the apparatus comprises:

a receiving module, configured to receive a first invocation request for the first smart contract,

wherein the first invocation request comprises the first block in the source blockchain; and

a processing module, configured to verify authenticity of the first block based on the first smart contract, wherein

the processing module is further configured to store the first block in the target blockchain if authenticity verification of the first block succeeds.

12. The apparatus according to claim 11, wherein the target blockchain stores a second block migrated from the source blockchain, and the second block is a previous block of the first block in the source blockchain; and

the processing module is further configured to determine, based on the first smart contract, whether the first block is a successor block of the second block, to verify the authenticity of the first block.

13. The apparatus according to claim 11 or 12, wherein the first smart contract records a maximum value that is of a block hash value and that is agreed on by a blockchain node in the source blockchain; and

the processing module is further configured to determine, based on the first smart contract, whether a block hash value of the first block is less than the maximum value of the block hash value, to verify the authenticity of the first block.

14. The apparatus according to any one of claims 11 to 13, wherein a blockchain node that stores the first block in the source blockchain is a first blockchain node, the first block is a block to which a digital signature is added by the blockchain node of the first block, and the first smart contract records a public key of the first blockchain node; and

the processing module is further configured to determine, based on the first smart contract and the public key of the first blockchain node, whether the digital signature of the first block is a digital signature of the first blockchain node, to verify the authenticity of the first block.

15. The apparatus according to any one of claims 11 to 14, wherein the first smart contract records first transaction information of a third block stored in a second blockchain node in the source blockchain, and a block height of the third block in the source blockchain is the same as a block height of the first block; and

the processing module is further configured to compare transaction information recorded in the first block and the first transaction information based on the first smart contract, to verify the authenticity of the first block.

16. The apparatus according to any one of claims 11 to 15, wherein the receiving module is further configured to:

receive a second invocation request for the first smart contract from a user node of the target

blockchain, wherein the second invocation request is used to query a block that is in the source blockchain and that is stored in the target blockchain.

17. The apparatus according to any one of claims 11 to 16, wherein the receiving module is further configured to receive the first invocation request for the first smart contract from a forwarding device, wherein the forwarding device is configured to migrate the first block from the source blockchain to the target blockchain.

18. The apparatus according to claim 17, wherein the forwarding device is one of a plurality of forwarding devices, and the plurality of forwarding devices are configured to migrate the first block from the source blockchain to the target blockchain.

19. The apparatus according to any one of claims 11 to 18, wherein the processing module is further configured to end an invocation process of the first smart contract if authenticity verification of the first block fails.

20. An apparatus for migrating a block, wherein the block is a first block that is in a source blockchain and that is to be migrated to a target blockchain, the target blockchain stores a first smart contract, the first smart contract is used to manage a block that is in the source blockchain and that is forwarded by a forwarding device, and the apparatus is applied to the forwarding device; and the apparatus comprises:

an obtaining module, configured to obtain the first block from the source blockchain; and

a sending module, configured to send a first invocation request for the first smart contract to the target blockchain node, wherein the first invocation request comprises the first block.

21. An apparatus for migrating a block, comprising a memory and a processor, wherein the memory stores executable code, and the processor is configured to execute the executable code, to implement the method according to any one of claims 1 to 9.

22. An apparatus for migrating a block, comprising a memory and a processor, wherein the memory stores executable code, and the processor is configured to execute the executable code, to implement the method according to claim 10.