Figure 1

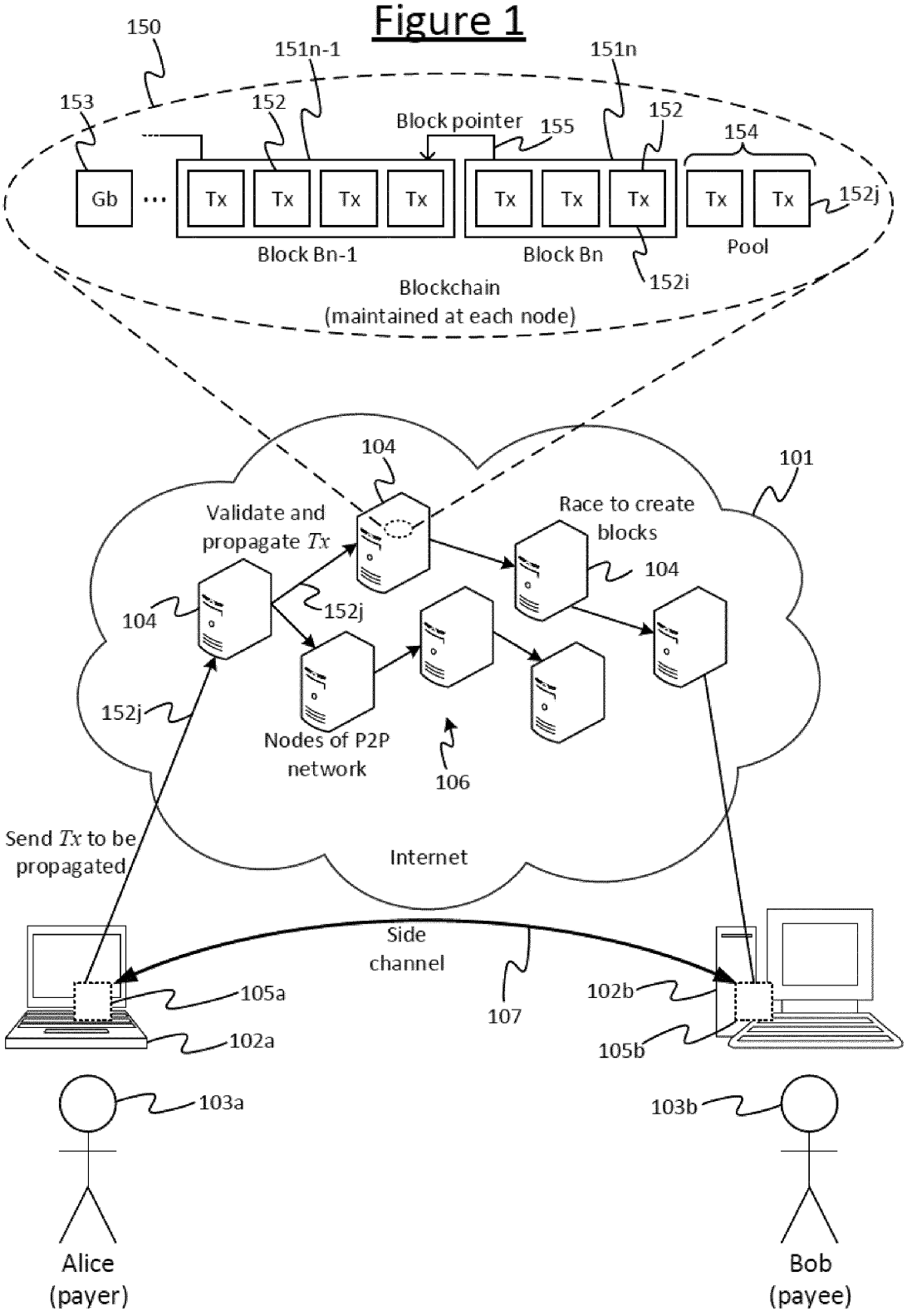
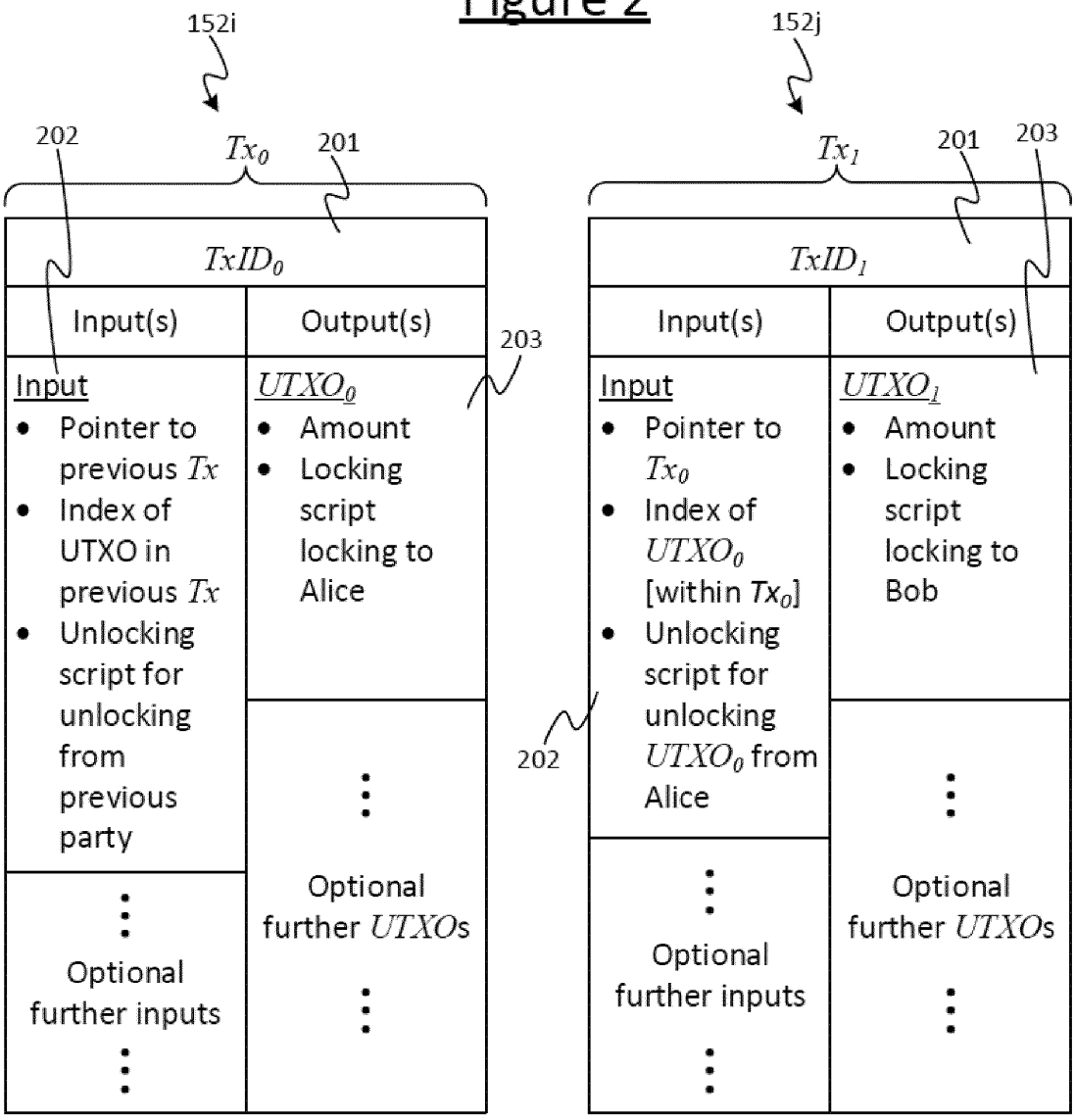

150
151n-1
151n
153
152
Block pointer
155
152
154
Gb
Tx
Tx
Tx
Tx
Tx
Tx
Tx
Tx
Tx
152j
Block Bn-1
Block Bn
Pool
152i
Blockchain
(maintained at each node)
104
101
Validate and
propagate Tx
Race to create
blocks
104
104
152j
152j
Nodes of P2P
network
106
Send Tx to be
propagated
Internet
Side
channel
107
105a
102a
102b
105b
103a
103b
Alice
(payer)
Bob
(payee)

Figure 2



Transaction from Alice to Bob

Validated by running: Locking script from output 203 of Tx_0, together with Alice's unlocking script from input 202 of Tx_1. This checks that Alice's unlocking script in Tx_1 meets the condition(s) defined in the locking script of previous transaction Tx_0.

Figure 3

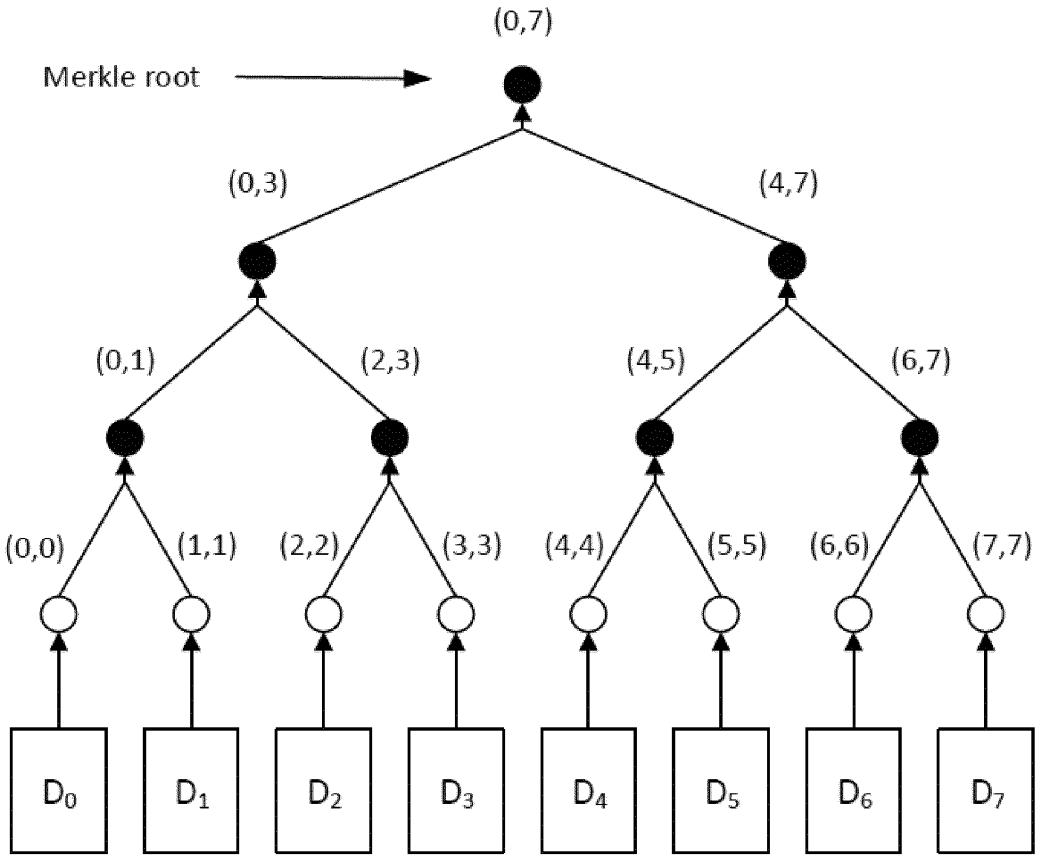

(0,7)
Merkle root
(0,3)
(4,7)
(0,1)
(2,3)
(4,5)
(6,7)
(0,0)
(1,1)
(2,2)
(3,3)
(4,4)
(5,5)
(6,6)
(7,7)
D0
D1
D2
D3
D4
D5
D6
D7

Figure 4

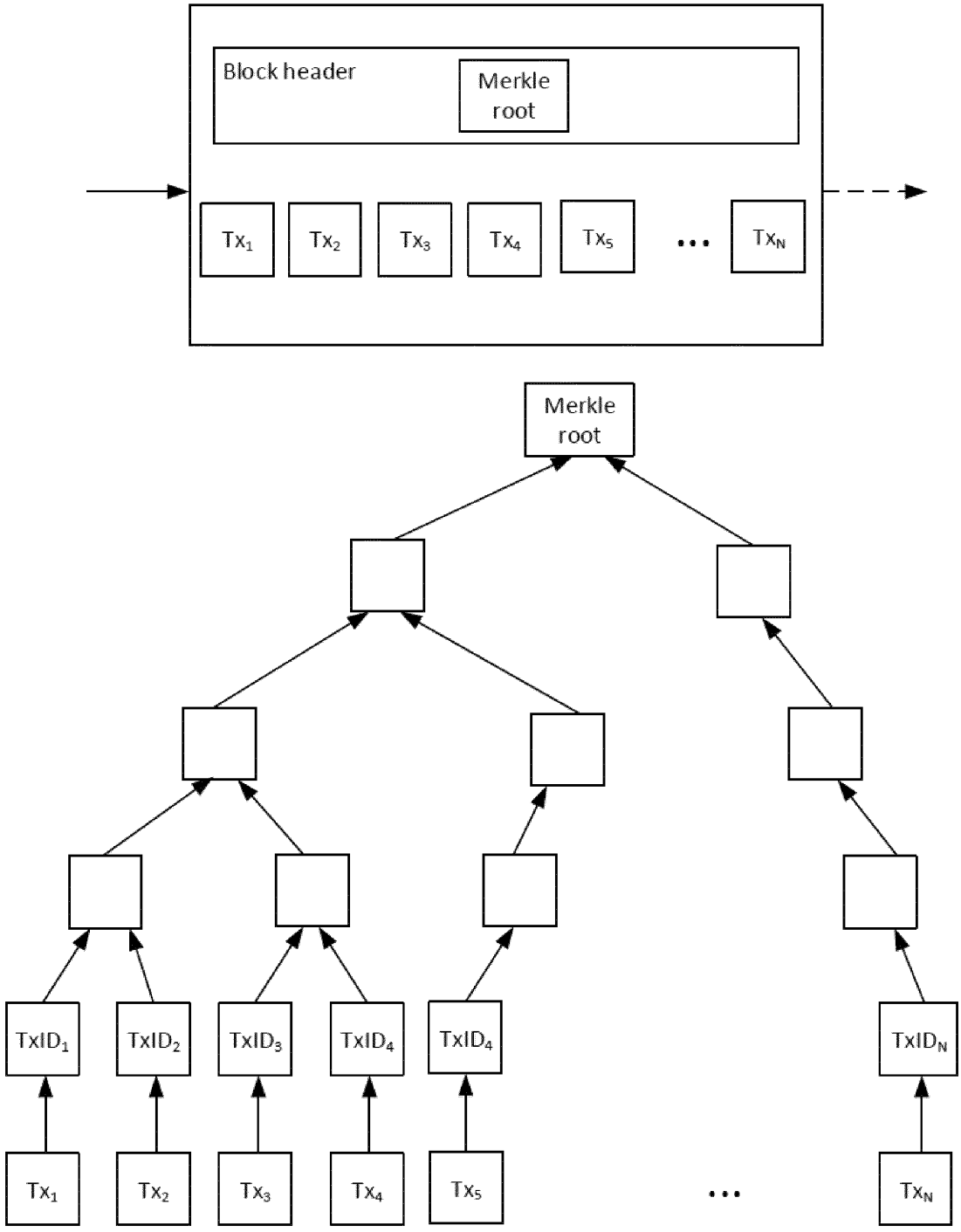

Block header
Merkle root
Tx1
Tx2
Tx3
Tx4
Tx5
...
TxN
Merkle root
TxID1
TxID2
TxID3
TxID4
TxID4
TxIDN
Tx1
Tx2
Tx3
Tx4
Tx5
...
TxN

Figure 5

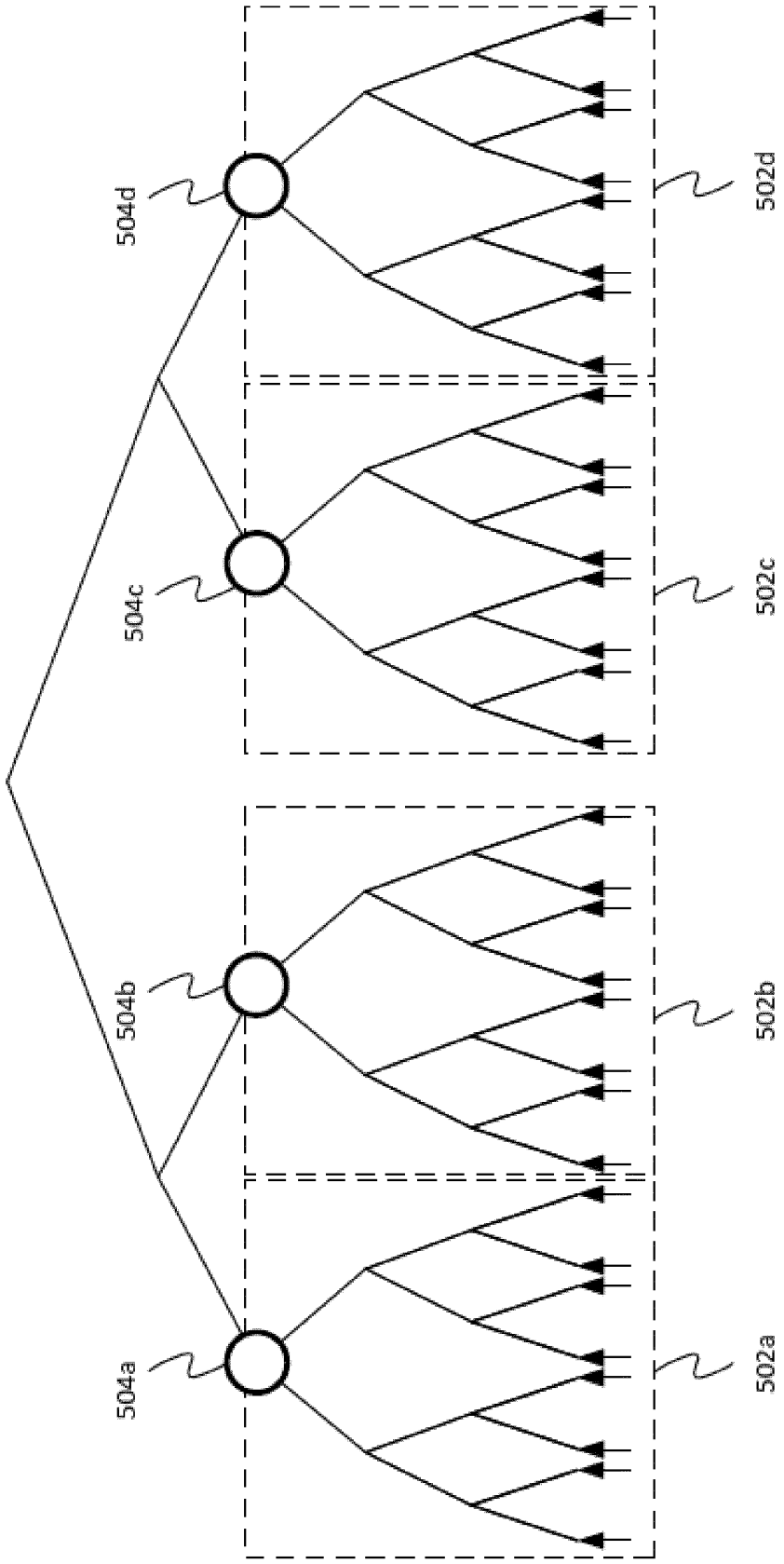


Figure 6

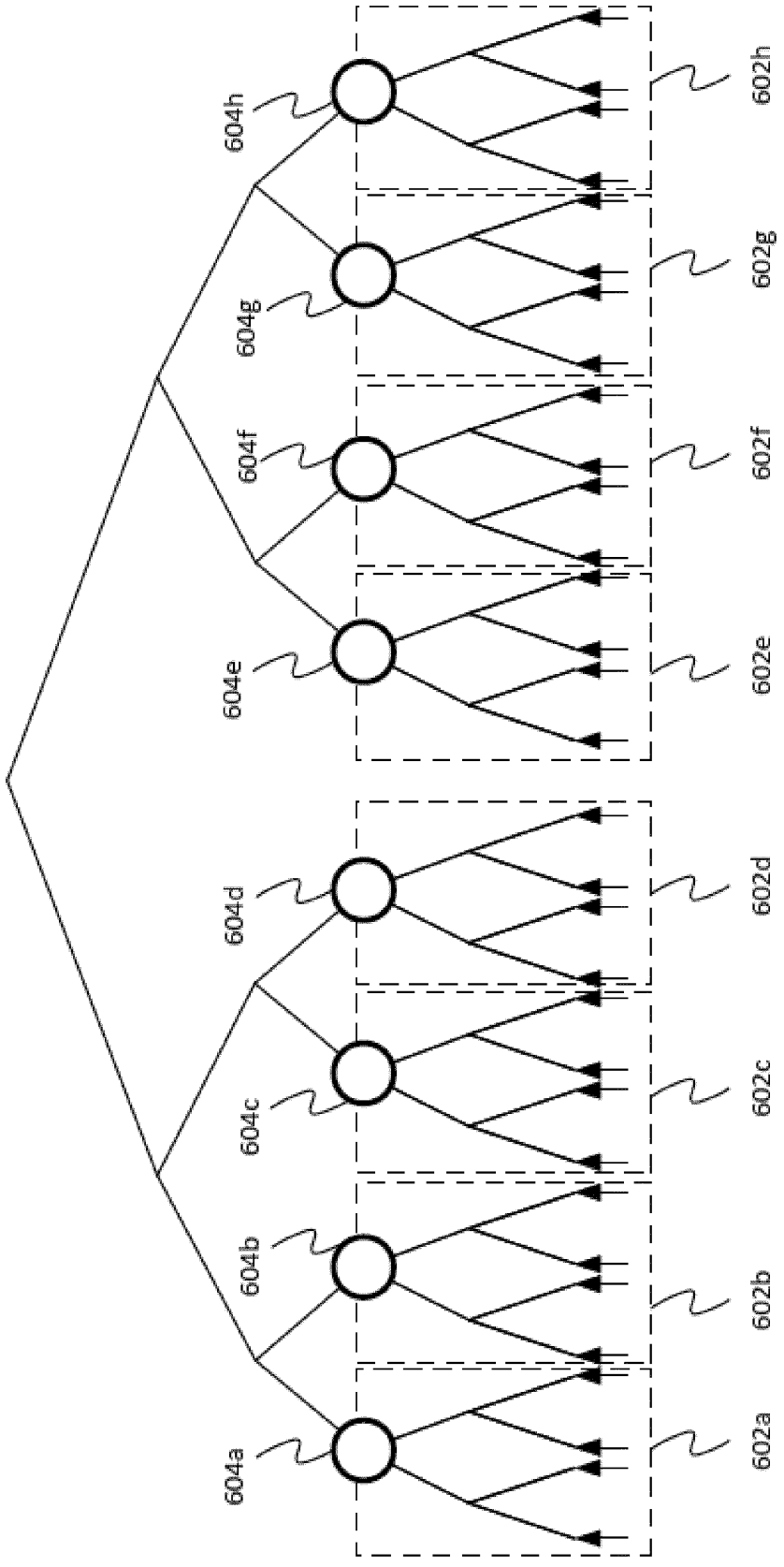

604h
602h
604g
602g
604f
602f
604e
602e
604d
602d
604c
602c
604b
602b
604a
602a

Figure 7

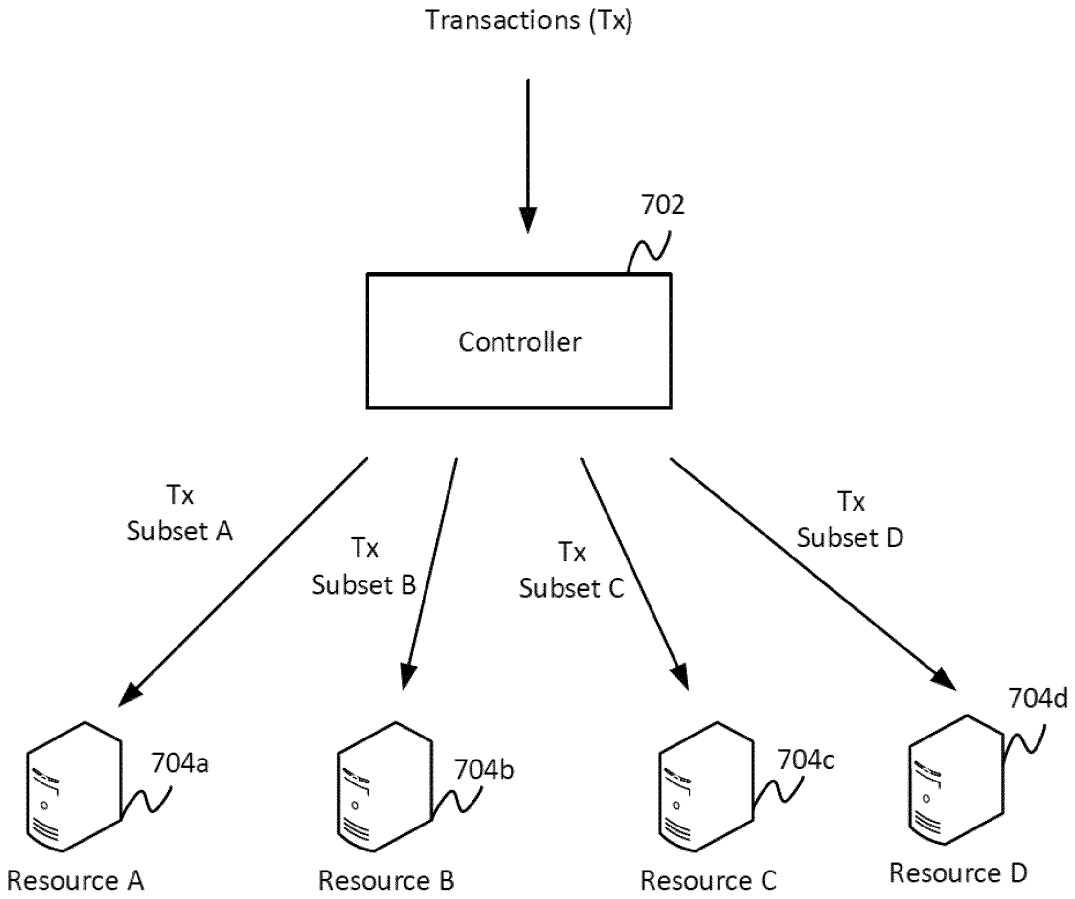

Transactions (Tx)
702
Controller
Tx
Subset A
Tx
Subset B
Tx
Subset C
Tx
Subset D
704a
704b
704c
704d
Resource A
Resource B
Resource C
Resource D

Figure 8

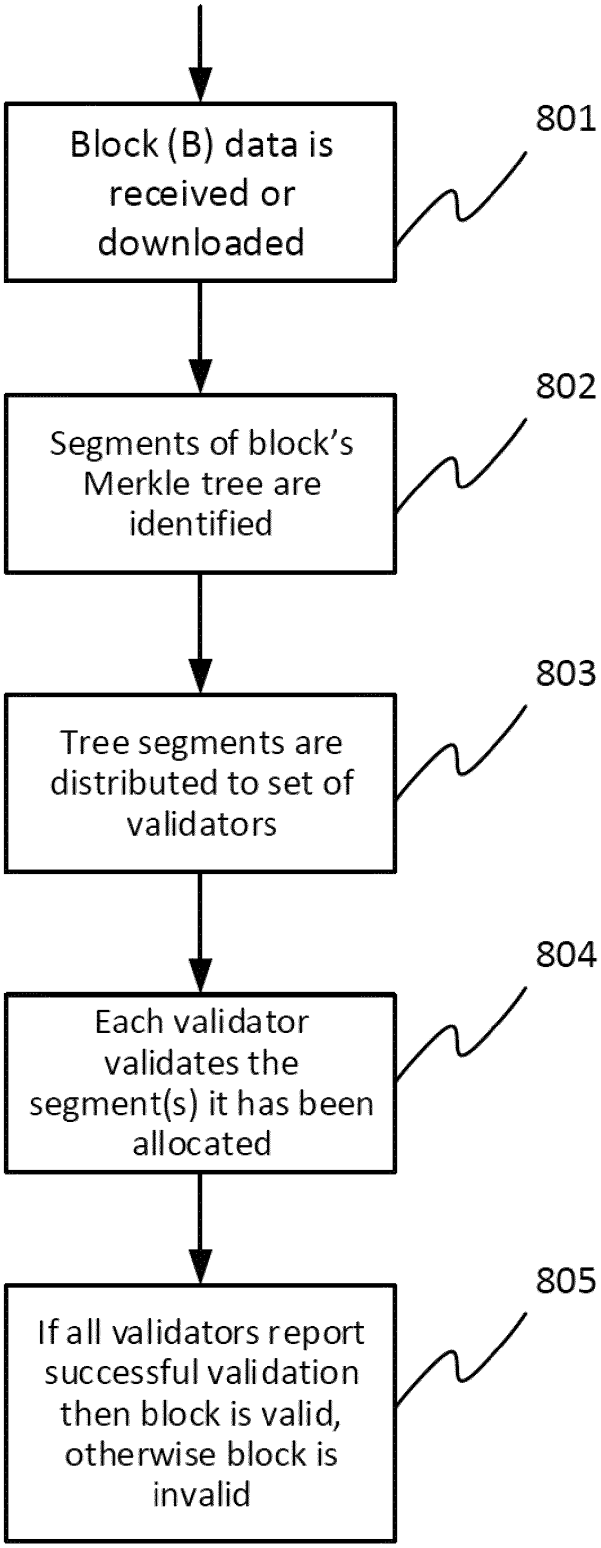

Block (B) data is received or downloaded
801
Segments of block's Merkle tree are identified
802
Tree segments are distributed to set of validators
803
Each validator validates the segment(s) it has been allocated
804
If all validators report successful validation then block is valid, otherwise block is invalid
805

Figure 9

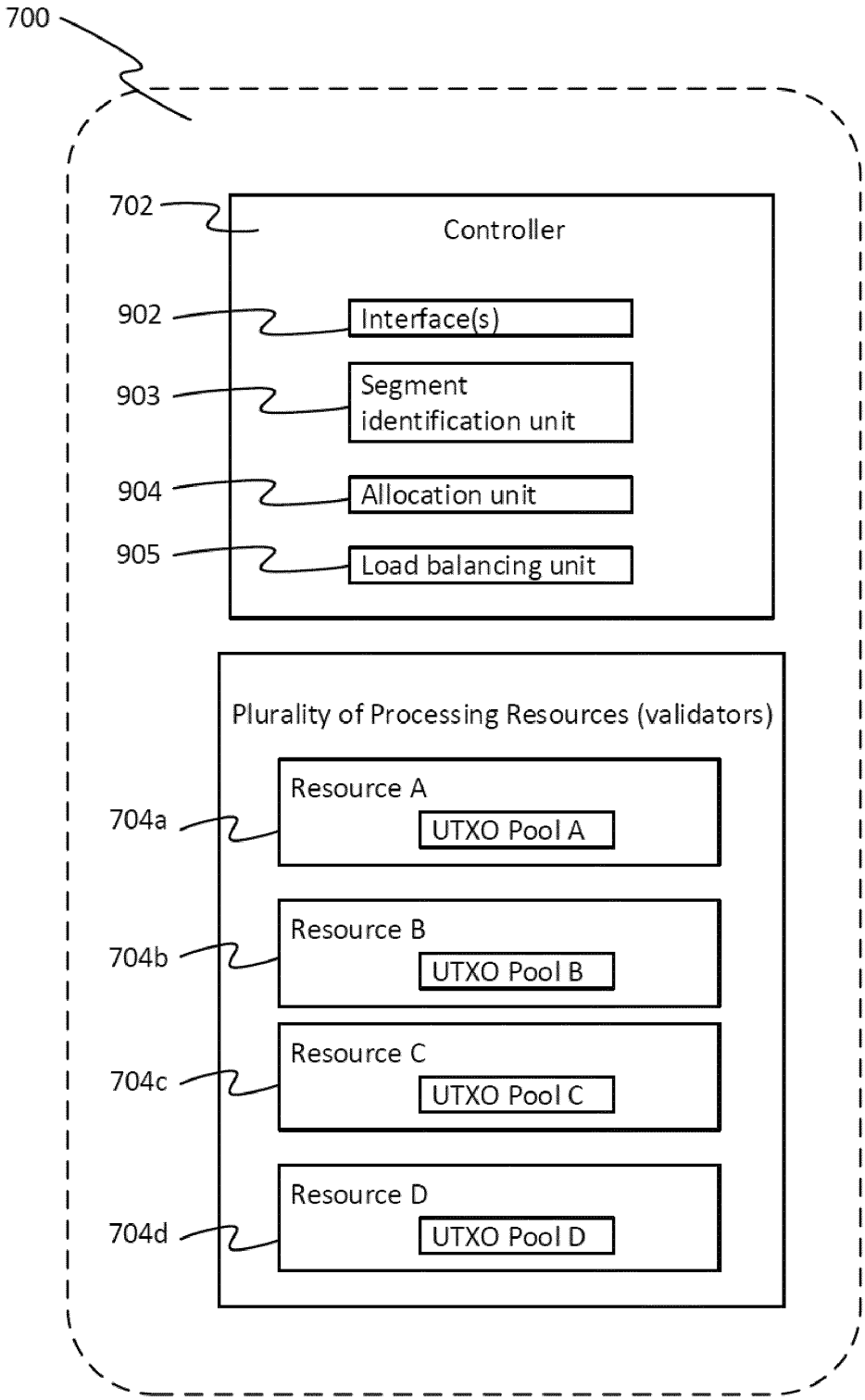

700
702
Controller
902
Interface(s)
903
Segment identification unit
904
Allocation unit
905
Load balancing unit
Plurality of Processing Resources (validators)
704a
Resource A
UTXO Pool A
704b
Resource B
UTXO Pool B
704c
Resource C
UTXO Pool C
704d
Resource D
UTXO Pool D

Figure 10

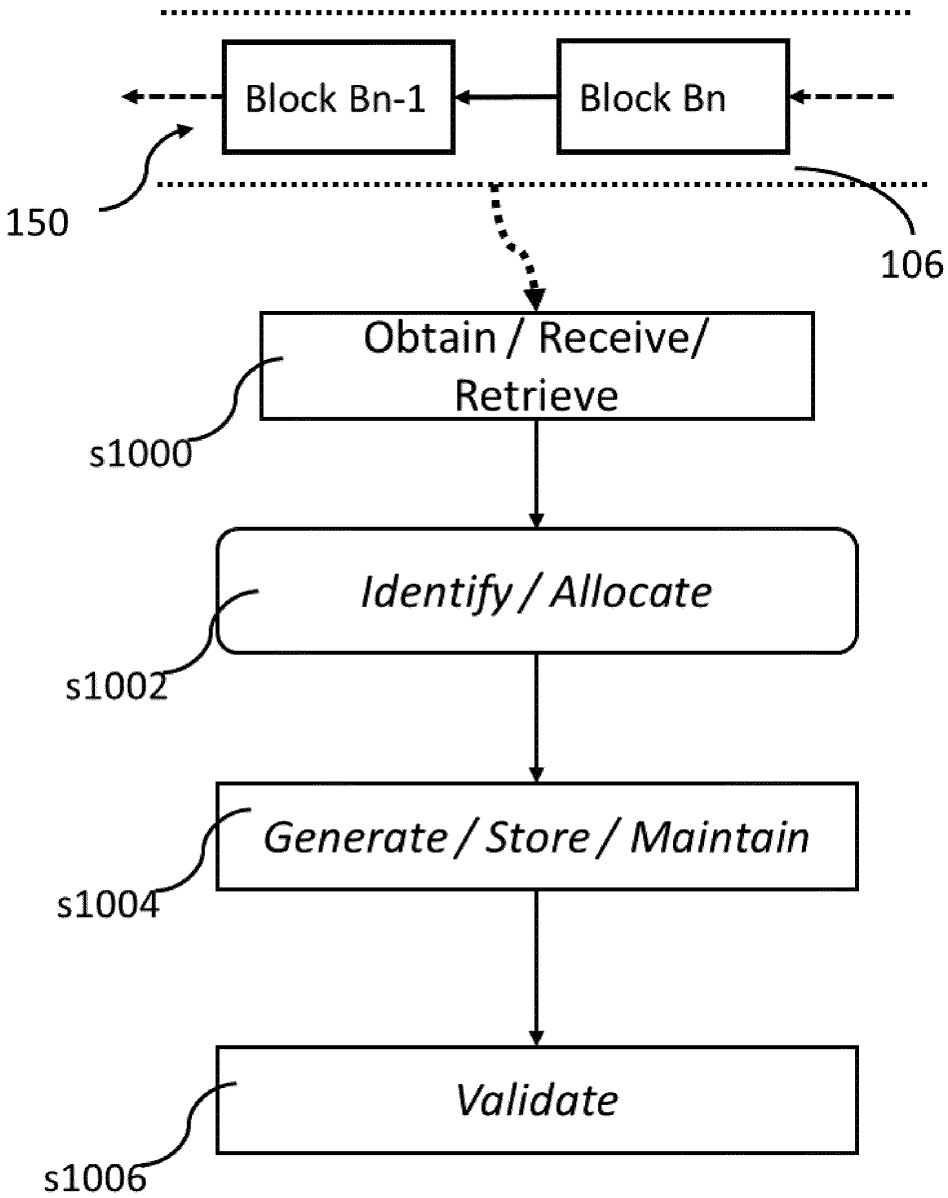

Block Bn-1
Block Bn
150
106
Obtain / Receive/ Retrieve
s1000
Identify / Allocate
s1002
Generate / Store / Maintain
s1004
Validate
s1006

Figure 11

Binary value	Allocated resource
000	1
001	2
010	3
011	4
100	5
101	6
110	7
111	8

(a)

HEX value	Allocated resource
0	1
1	2
2	3
3	4
4	5
5	6
6	7
7	8
8	9
9	10
a	11
b	12
c	13
d	14
e	15
f	16

(b)

Binary value	Allocated resource
0000	1
0001	2
0010	3
0011	4
0100	5
0101	6
0110	7
0111	8
1000	9
1001	10
1010	11
1011	12
1100	13
1101	14
1110	15
1111	16

(c)

HEX value range	Allocated resource
00 - 0f	1
10 - 1f	2
20 - 2f	3
30 - 3f	4
40 - 4f	5
50 - 5f	6
60 - 6f	7
70 - 7f	8
80 - 8f	9
90 - 9f	10
a0 - af	11
b0 - bf	12
c0 - cf	13
d0 - df	14
e0 - ef	15
f0 - ff	16

(d)

Figure 12

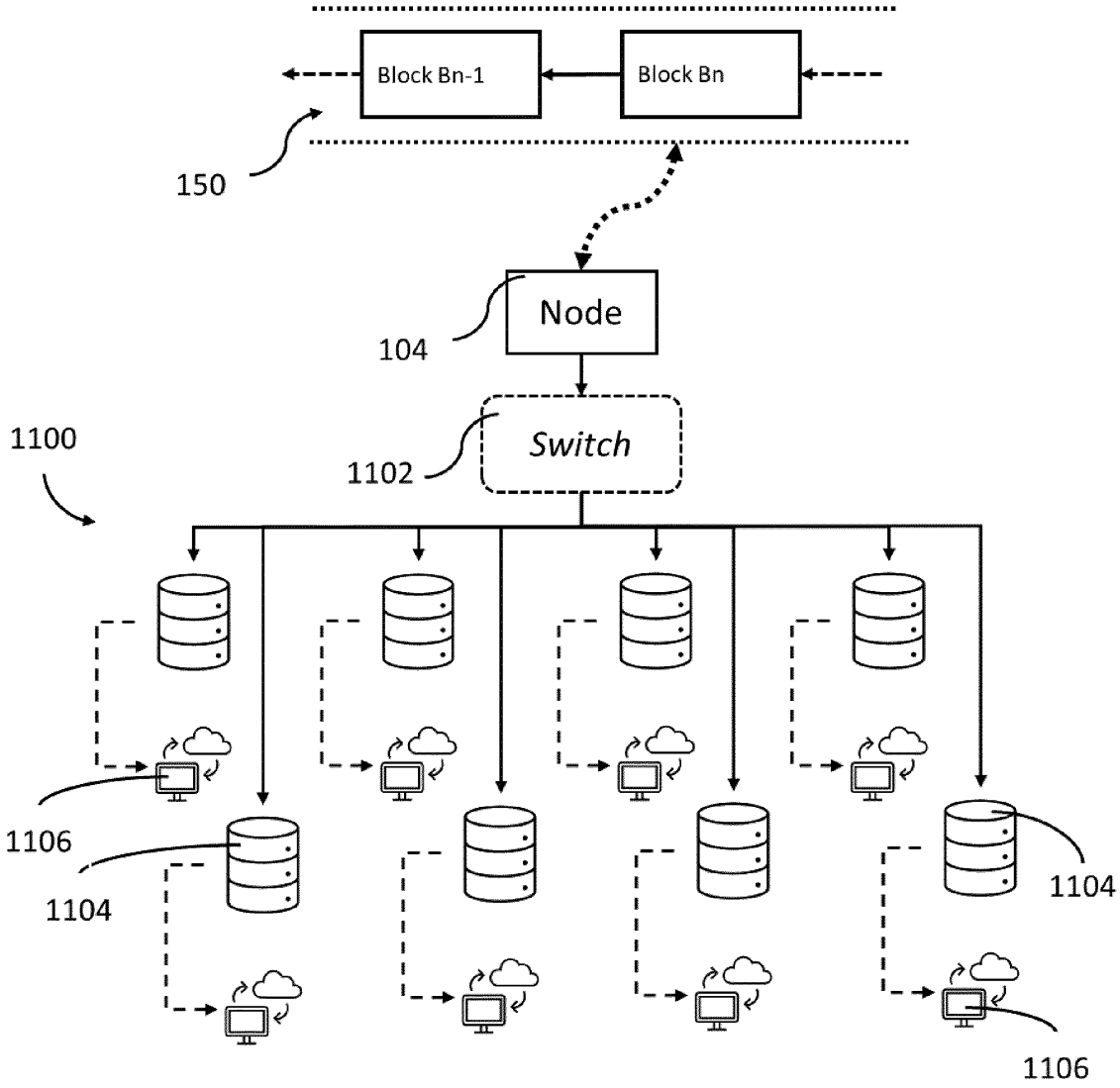

Block Bn-1
Block Bn
150
Node
104
Switch
1102
1100
1106
1104
1104
1106

Figure 13

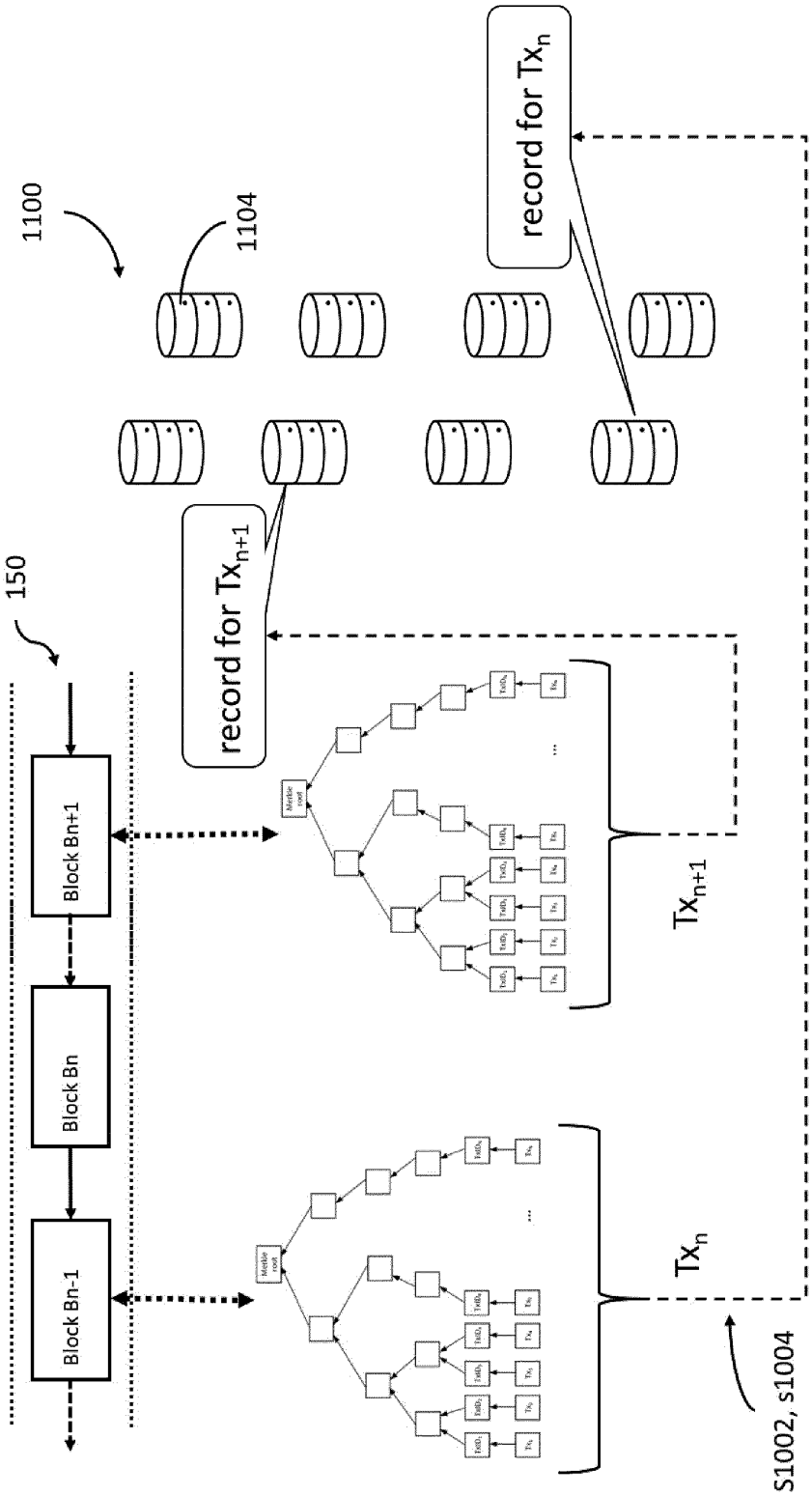

1100
1104
record for Tx_n
record for Tx_{n+1}
150
Block Bn+1
Block Bn
Block Bn-1
Tx_{n+1}
Tx_n
S1002, s1004

Figure 14

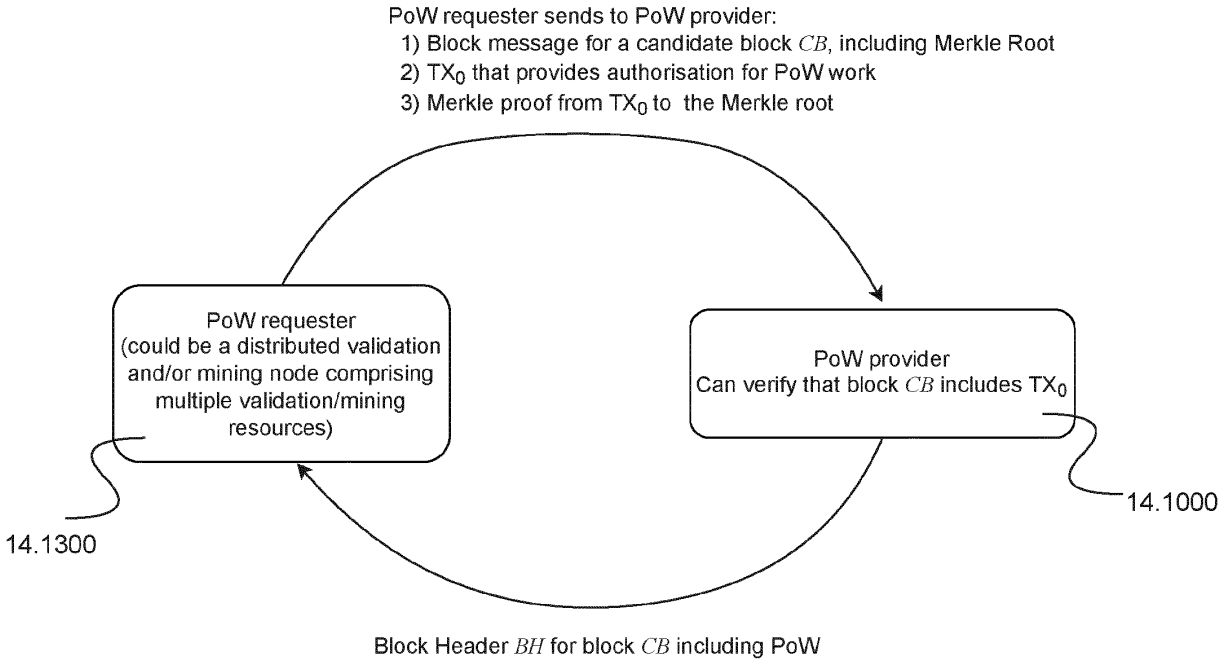

PoW requester sends to PoW provider:
1) Block message for a candidate block CB, including Merkle Root
2) TX0 that provides authorisation for PoW work
3) Merkle proof from TX0 to the Merkle root
PoW requester
(could be a distributed validation and/or mining node comprising multiple validation/mining resources)
PoW provider
Can verify that block CB includes TX0
14.1300
14.1000
Block Header BH for block CB including PoW

Figure 15 *(prior art - block mining process)*

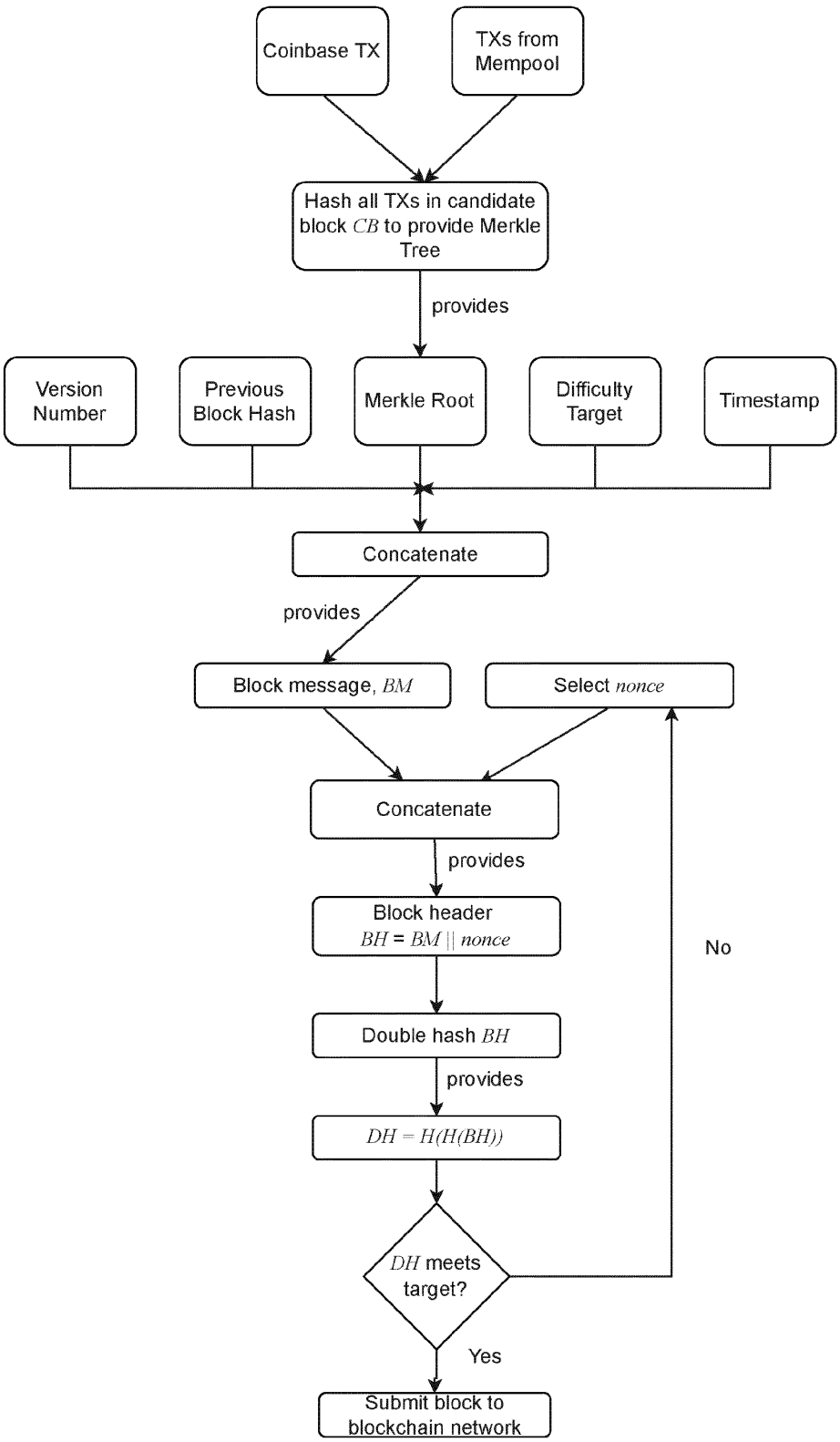


Figure 16 *(disclosure compared to prior art process)*

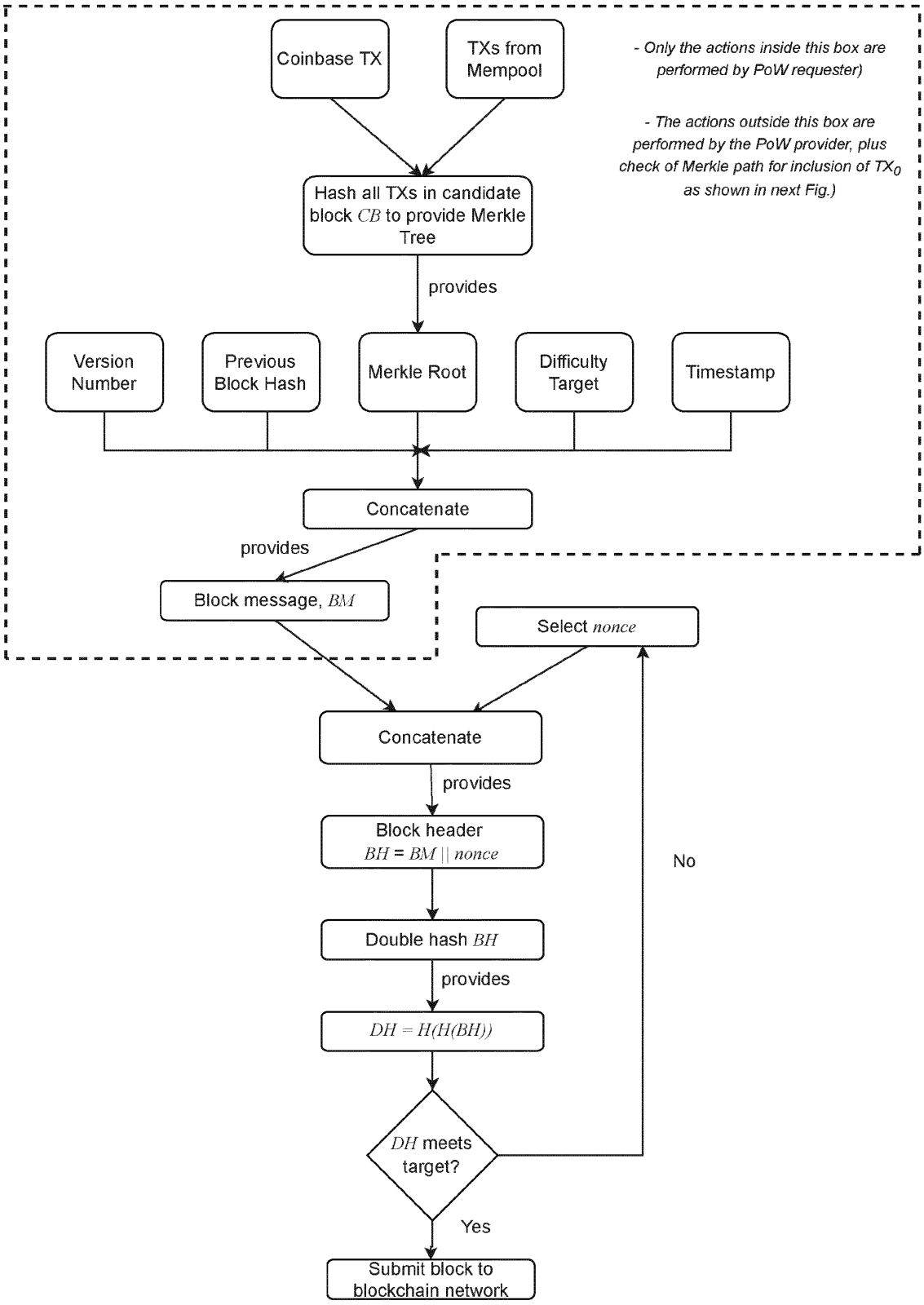


Figure 17

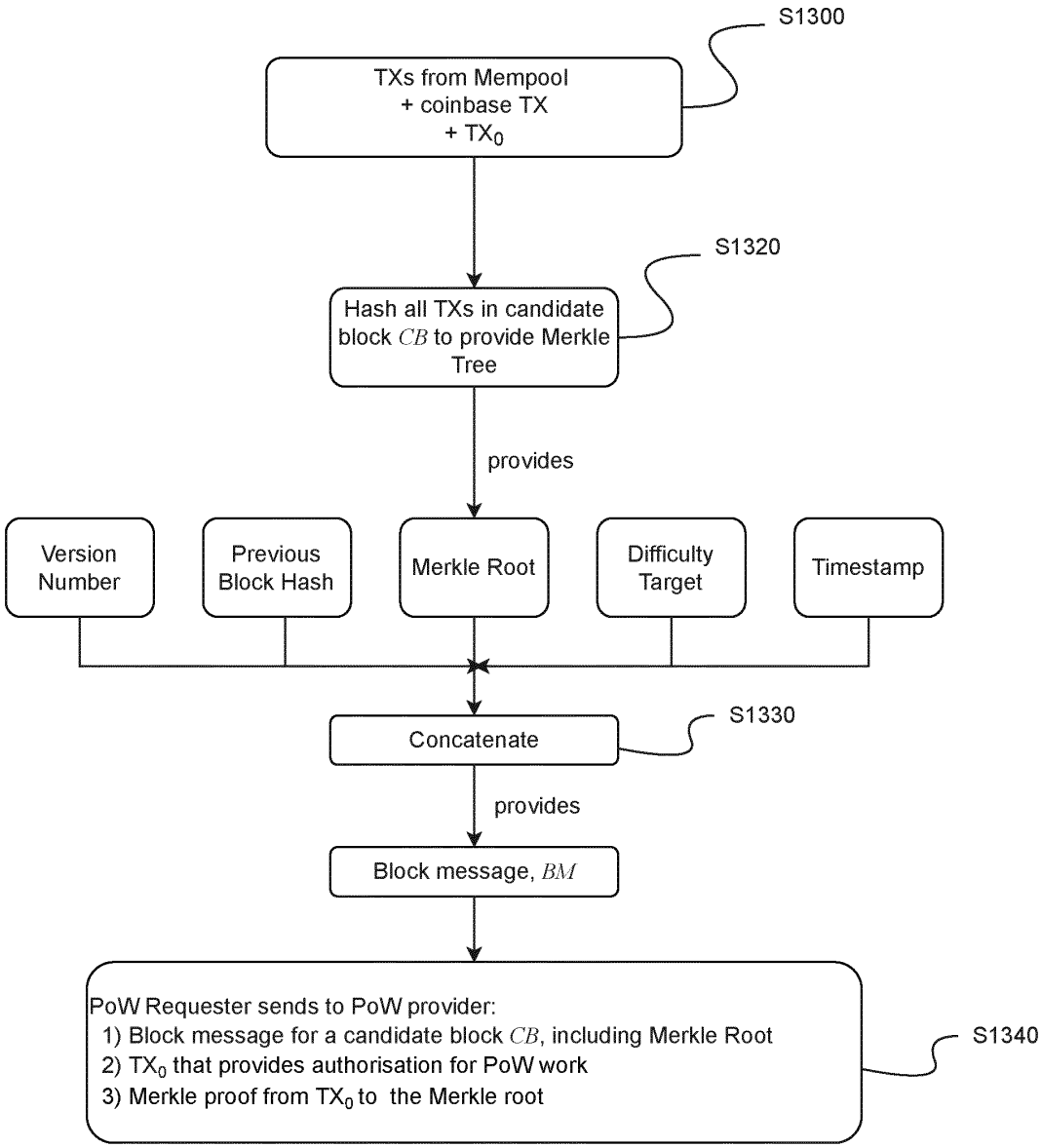
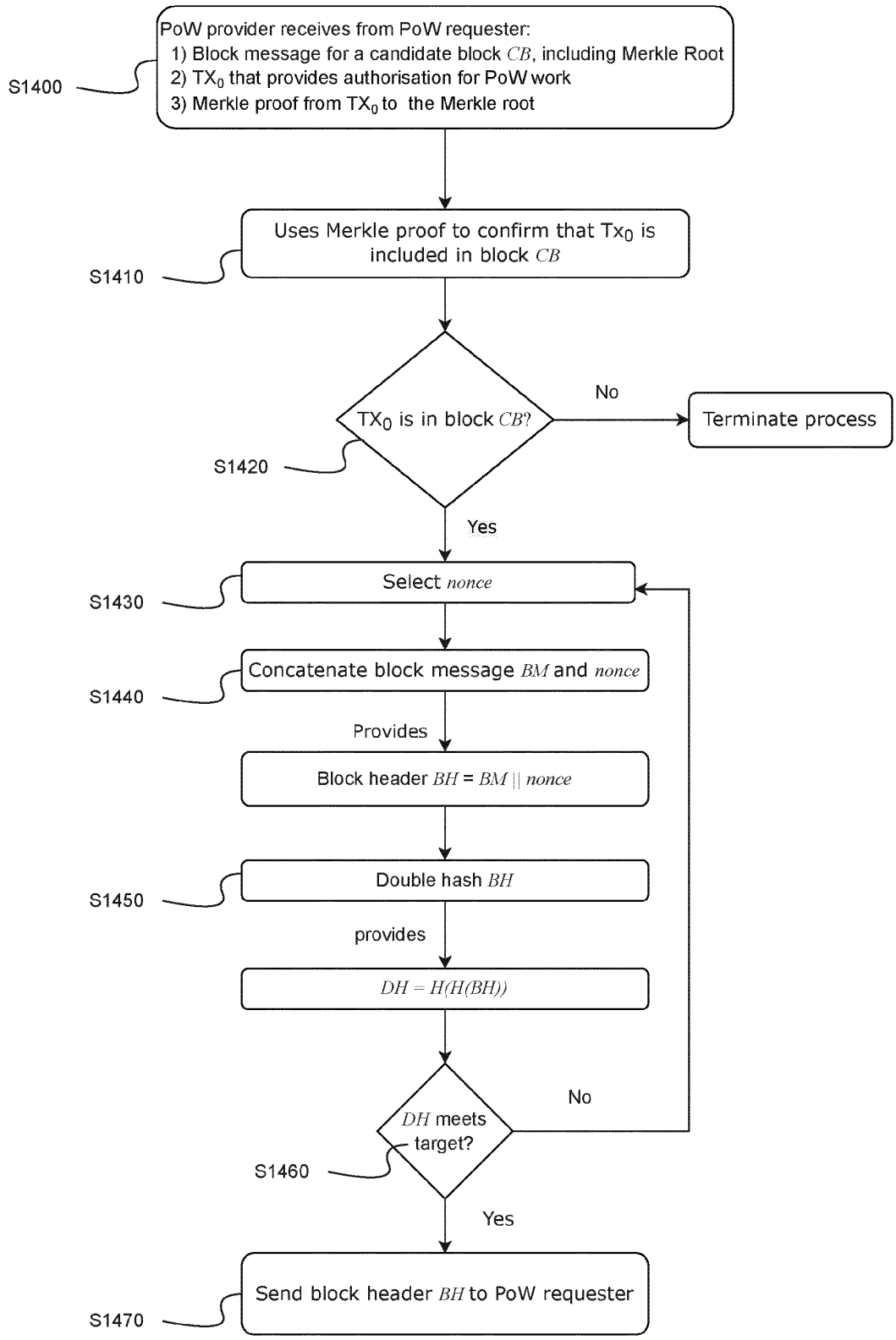

S1300
TXs from Mempool
+ coinbase TX
+ TX_0
S1320
Hash all TXs in candidate block CB to provide Merkle Tree
provides
Version Number
Previous Block Hash
Merkle Root
Difficulty Target
Timestamp
S1330
Concatenate
provides
Block message, BM
PoW Requester sends to PoW provider:
1) Block message for a candidate block CB, including Merkle Root
2) TX_0 that provides authorisation for PoW work
3) Merkle proof from TX_0 to the Merkle root
S1340

Figure 18


S1400
PoW provider receives from PoW requester:
1) Block message for a candidate block CB, including Merkle Root
2) TX0 that provides authorisation for PoW work
3) Merkle proof from TX0 to the Merkle root
S1410
Uses Merkle proof to confirm that Tx0 is included in block CB
S1420
TX0 is in block CB?
No
Terminate process
Yes
S1430
Select nonce
S1440
Concatenate block message BM and nonce
Provides
Block header BH = BM || nonce
S1450
Double hash BH
provides
DH = H(H(BH))
S1460
DH meets target?
No
Yes
S1470
Send block header BH to PoW requester

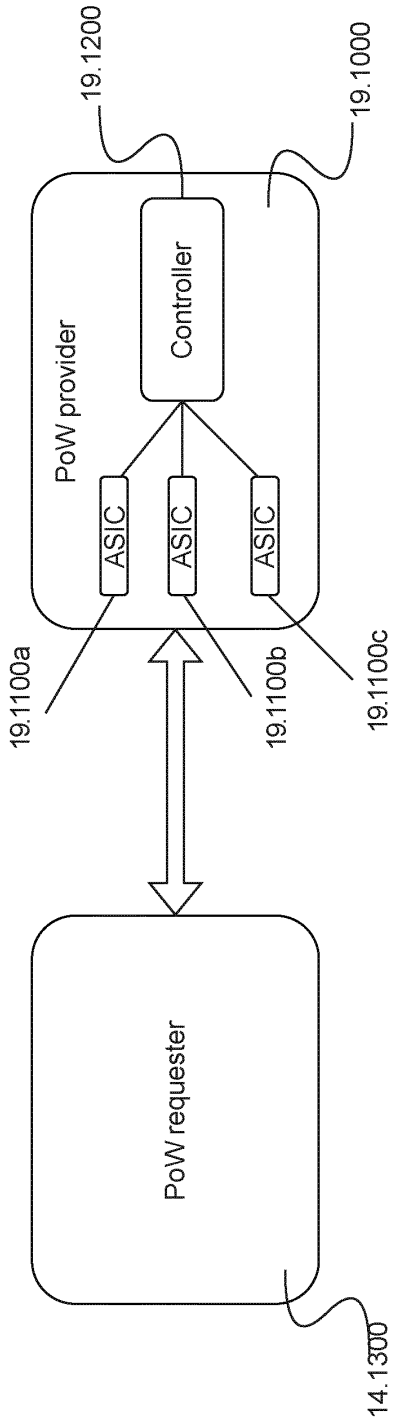
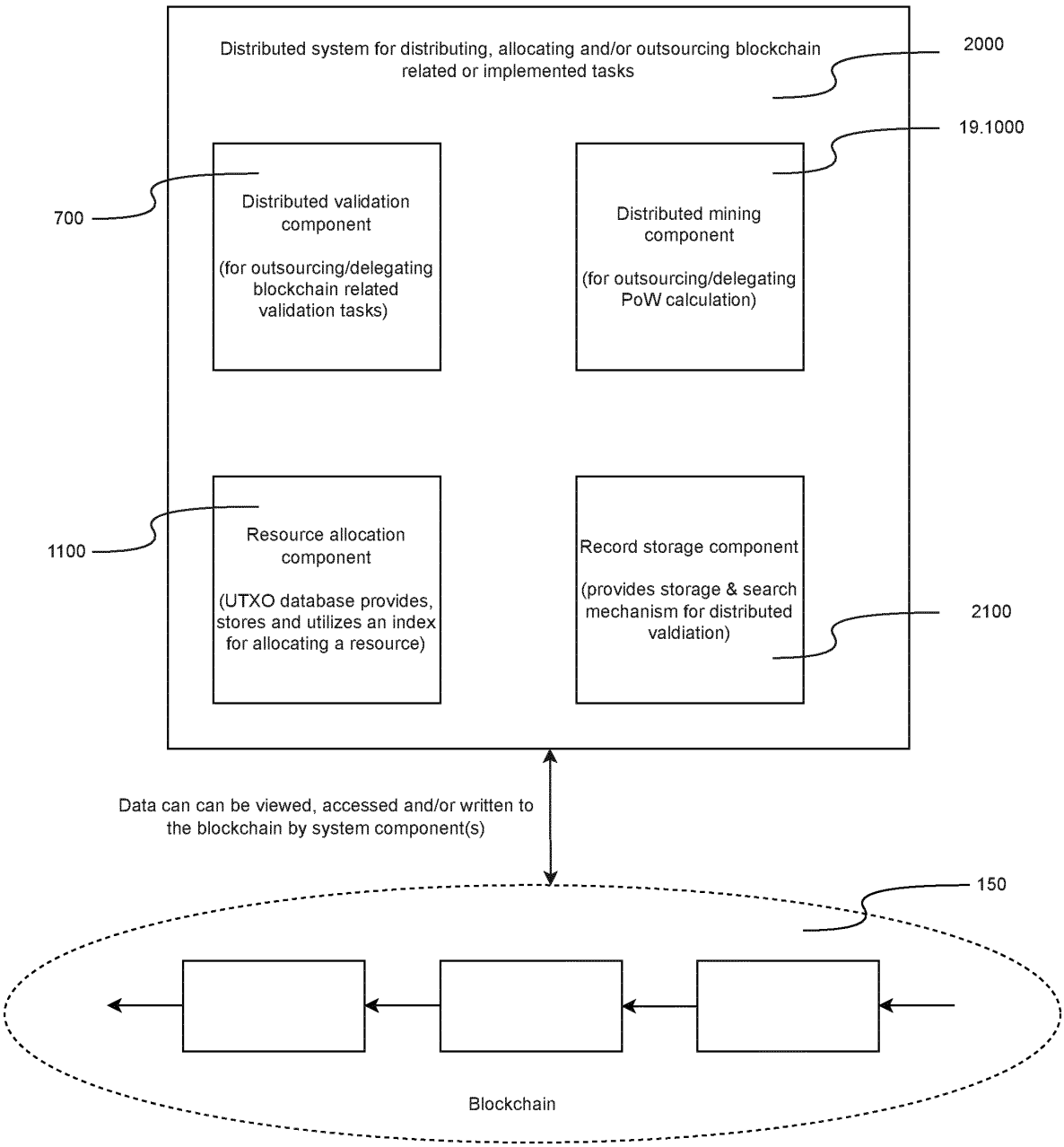

19.1200
19.1000
PoW provider
Controller
ASIC
ASIC
ASIC
19.1100a
19.1100b
19.1100c
PoW requester
14.1300

Figure 19

FIGURE 20


Distributed system for distributing, allocating and/or outsourcing blockchain related or implemented tasks
2000
19.1000
700
Distributed validation component
(for outsourcing/delegating blockchain related validation tasks)
Distributed mining component
(for outsourcing/delegating PoW calculation)
1100
Resource allocation component
(UTXO database provides, stores and utilizes an index for allocating a resource)
Record storage component
(provides storage & search mechanism for distributed valdiation)
2100
Data can can be viewed, accessed and/or written to the blockchain by system component(s)
150
Blockchain