We claim:

1. A computer-implemented method in a computer network, the computer-implemented method comprising:

configuring a distributed electronic ledger in the electronic memory of one or more computers in a blockchain system, the distributed electronic ledger having a plurality of backward-linked interconnected blocks, arranged as one or more instances of linear blockchain data structures, non-linear block arrangements, n-dimensional mesh or lattice data structures, or directed acyclic graphs;

configuring each of the interconnected blocks to include an ordered set of individual data records, such that at least one of the records reflects a transformation of at least a portion of a global state of the distributed electronic ledger;

configuring the blockchain system to comprise a peer-to-peer network of computer nodes, with one or more computers configured as wallet nodes, and with one or more computers configured as block-building nodes;

configuring the one or more wallet nodes to transmit one or more of the data records to one or more block-building nodes in the peer-to-peer network;

configuring the one or more block-building nodes to construct one or more interconnected blocks in the distributed electronic ledger by selecting and ordering one or more of the data records to be included in said one or more new interconnected blocks; and

implementing in the blockchain system, a trust-but-verify data propagation process, wherein at least one block-building node is configured to create at least one new interconnected block in the blockchain system before verifying correctness of one or more preceding blocks to which said at least one new interconnected block is linked.

2. The computer-implemented method of Claim 1, wherein the trust-but-verify data propagation process further includes propagating said at least one new interconnected block in the blockchain system via a gossip protocol.

3. The computer-implemented method of Claim 2, wherein the gossip protocol enables said at least one new interconnected block to be propagated across the blockchain system, further including:

transmitting said at least one new interconnected block via the gossip protocol as a block header, while excluding the bulk of the data belonging to said at least one interconnected block.

4. The computer-implemented method of Claim 2, wherein the gossip protocol enables said at least one new interconnected block to be transmitted without one or more data records belonging to said at least one new interconnected block, such that said one or more data records are transmitted separately from said at least one new interconnected block.

5. The computer-implemented method of Claim 1, further including:

at least one other block-building node creating at least one next interconnected block on top of said at least one new interconnected block after said at least one new interconnected block is distributed to the blockchain system; and

the at least one other block-building node simultaneously creating the at least one next interconnected block while one or more data records belonging to said at least one new interconnected block are downloaded by the at least one other block-building node, or while said at least one new interconnected block is validated.

6. The computer-implemented method of Claim 1, wherein the trust-but-verify data propagation process further includes causing the at least one block-building node to include a Bloom filter in a block header of said at least one new interconnected block in the blockchain system, such that the trust-but-verify data propagation process causes the at least one block-building node to distribute the Bloom filter with the at least one new interconnected block in the blockchain system.

7. The computer-implemented method of Claim 6, wherein the Bloom filter is configured to represent a set of distributed electronic ledger accounts or other data affected or operated upon during one of (i) the creation, (ii) the validation, or (iii) the execution of said at least one new interconnected block in the blockchain system

8. The computer-implemented method of Claim 6 wherein the Bloom filter of a block is a data structure configured at least

with the property that any account or other data found in the Bloom filter is possible or likely to have its state modified by the validation and execution of the block, or to have been operated upon or otherwise affected by the block or by data records belonging to the block, and

with the property that any account not found in the Bloom filter is certain to be unchanged as to its state or unaffected by the block or by data records belonging to the block.

9. The computer-implemented method of Claim 8, wherein creating said at least one new interconnected block further includes:

identifying accounts affected or operated upon when one or more smart contracts execute, or when one or more data records are added to the said at least one new interconnected block;

checking to ensure that each account identified is not contained within a Bloom filter of at least one prior interconnected block that precedes said at least one new interconnected block; and

if an identified account is contained within a Bloom filter of the at least one prior interconnected block, halting execution of said one or more smart contracts or halting inclusion of said one or more data records, and discarding said one or more smart contract or one or more data records from said at least one new interconnected block.

10. The computer-implemented method of Claim 1, wherein the trust-but-verify data propagation process uses the directed acyclic graphs (DAG) data structure for arranging said at least one new interconnected block.

11. The computer-implemented method of Claim 10, wherein the trust-but-verify data propagation process allows said at least one new interconnected block to have back references to one or more interconnected blocks or sub-chains in the blockchain system, causing at least a portion of the distributed electronic ledger to constitute a directed acyclic graph (DAG).

12. The computer-implemented method of Claim 11, wherein the trust-but-verify data propagation process ensures that the multiple interconnected blocks or sub-chains being built upon simultaneously by said new interconnected block include records associated with distinct, mutually exclusive sets of accounts or other data, such that the multiple interconnected blocks or sub-chains being built upon affect mutually-exclusive sets of accounts or other data.

13. The computer-implemented method of Claim 12, wherein the trust-but-verify data propagation process enables the multiple interconnected blocks or sub-chains to be quickly compared to determine if any of the multiple interconnected blocks or sub-chains operate upon common accounts or other data.

14. The computer-implemented method of Claim 12, further including comparing respective Bloom filters of the multiple interconnected blocks or sub-chains in order to determine with certainty that interconnected blocks or sub-chains in the blockchain system affect or operate upon mutually-exclusive sets of accounts or other data.

15. The computer-implemented method of Claim 1, wherein the trust-but-verify data propagation process enables simultaneous creation of a plurality new interconnected blocks in the distributed electronic ledger, such that multiple nodes in the blockchain system are creating one or more of the plurality of new interconnected blocks in parallel; and

wherein the trust-but-verify data propagation process ensures that data records are not propagated widely across the blockchain system prior to interconnected block creation, such that data records produced by a given node are only shared with certain other nodes within the blockchain system; and

wherein the trust-but-verify data propagation process ensures that a respective block-building node only creates said at least one new interconnected block using data records propagated by certain nodes within the blockchain system.

16. The computer-implemented method of Claim 1, wherein if a block-building node discovers that said at least one new interconnected block cannot be validated, said at least one new interconnected block is discarded along with any new or subsequent block which is built upon said at least one new interconnected block; and

wherein a block cannot be validated if one of the following occurs: (i) a Merkle tree root cannot be reproduced using a set of data records belonging to said block in the blockchain system, (ii) a block hash in said block is incorrectly specified, or (iii) a data record belonging to said block is invalid or violates a rule in the blockchain system.

17. The computer-implemented method of Claim 2, wherein the trust-but-verify data propagation process increases transaction throughput of the blockchain system relative to transaction throughput of a Nakamoto consensus proof-of-work blockchain system.

18. The computer-implemented method of Claim 2, wherein the at least one block-building node is an identifiable block-building node configured with a reward address that is included with at least one subject interconnected block being built by said at least one identifiable block-building node; and

wherein said at least one subject interconnected block is a signed block cryptographically signed by said at least on identifiable block-building node using at least one cryptographic key associated with said reward address,

wherein the at least one signed block is propagated in the blockchain network.

19. The computer-implemented method of Claim 18 wherein configuring the trust-but-verify data propagation process further includes at least one of:

(a) configuring the signed block to specify a build fee as a quantity of distributed electronic ledger tokens, which signed block is configured as an incentivized block, and at least one other block-building node generates a new subsequent block configured with a reference to said incentivized block;

wherein the incentivized block build fee is transferred from the reward address to a distributed ledger address specified by the at least one other block-building node; or

(b) configuring the signed block to specify tokens-at-risk as a quantity of distributed electronic ledger tokens, which signed block thereby functions as a guaranteed block, and configuring at least one other node in the blockchain network constructs a penalty record that contains or references the guaranteed block,

which penalty record, when incorporated into a subsequent new block, causes the specified tokens-at-risk to be transferred from the reward address to a distributed ledger address specified by the at least one other node in the blockchain network, if the guaranteed block is invalid.