

What is claimed is:

1. A method for managing transactions in a distributed database system including a plurality of nodes, the method including:

maintaining, at a first node of the plurality of nodes, records of a plurality of transactions, each transaction executing on one or more of the plurality of nodes, each record having a transaction state of a plurality of transaction states, the records including a record for a first transaction and a record for a second transaction, execution of the first transaction at a second node of the plurality of nodes including an operation for accessing a first data element stored on the second node and execution of the second transaction at the second node including an operation for accessing the first data element stored on the second node;

receiving, at the second node from the first node, a list of transactions that includes any of the plurality of transactions executing on the second node and having a first transaction state at a time of initiation of the second transaction, the list of transactions including the first transaction;

determining that an outcome of the second transaction depends on an outcome of the first transaction based at least in part on the list of transactions;
and

suspending, based on determining that the outcome of the second transaction depends on the outcome of the first transaction, execution of the second transaction; and

resuming the execution of the second transaction after the first transaction is completed.

2. The method of any preceding claim wherein determining that an outcome of the second transaction depends on an outcome of the first transaction based at least in part on the list of transactions includes determining that a time of initiation of the first transaction occurs prior to the time of initiation of the second transaction and that a commit time of the first transaction occurs prior to the time of initiation of the second transaction.
3. The method of any preceding claim wherein the list of transactions is received at the second node at the time of initiation of the second transaction.
4. The method of any preceding claim wherein the transactions included in the list of transactions consist of the transactions executing on the second node and having the first transaction state at the time of initiation of the second transaction.
5. The method of any preceding claim wherein the list of transactions includes, for each transaction in the list of transactions, a time of initiation of the transaction.
6. The method of any preceding claim wherein the first transaction state indicates that a transaction is preparing to complete.
7. The method of any preceding claim wherein the first transaction writes the first data element and the second transaction reads the first data element, and a version of the first data element read by the second transaction depends on the outcome of the first transaction.
8. The method of claim 7 wherein possible outcomes of the first transaction include a transaction aborted outcome and a transaction committed outcome.
9. The method of claim 8, wherein resuming execution of the second transaction after completion of the first transaction comprises reading a first version of the first data element if the outcome of the first transaction is the transaction aborted outcome.

10. The method of claim 8, wherein resuming execution of the second transaction after completion of the first transaction comprises reading a second, different version of the first data element written by the first transaction if the outcome of the first transaction is the transaction committed outcome.

11. The method of any preceding claim wherein one or both of the first transaction and the second transaction accesses data elements stored on a third node of the plurality of nodes.

12. The method of claim 1 wherein both the first transaction and the second transaction attempt to write the first data element and the second transaction is in the first transaction state.

13. The method of claim 12 wherein determining that an outcome of the second transaction depends on an outcome of the first transaction based at least in part on the list of transactions includes determining that a time of initiation of the second transaction occurs after a time of initiation of the first transaction and before a commit time of the first transaction.

14. The method of claim 12 wherein the first transaction is scheduled to commit its write before the second transaction and whether or not the second transaction is aborted depends on whether the first transaction results in a transaction aborted outcome or a transaction committed outcome.

15. The method of claim 14, wherein resuming execution of the second transaction after completion of the first transaction comprises writing a first value of the first data element if the outcome of the first transaction is the transaction aborted outcome.

16. The method of claim 14, wherein resuming execution of the second transaction after completion of the first transaction comprises aborting the second transaction if the outcome of the first transaction is the transaction committed outcome.

17. The method of claim 1, further comprising:

determining, based on data in the list of transactions received at the second node, whether the first transaction started before the second transaction;

wherein suspending the execution of the second transaction comprises
suspending execution of the second transaction in response to
determining that the first transaction started before the second transaction and determining that the outcome of the second transaction depends on the outcome of the first transaction.
18. The method of claim 1, wherein suspending the execution of the second transaction comprises suspending the execution of the second transaction to place the second transaction in a sleep state.
19. A computer-readable medium storing software in a non-transitory form, for managing transactions in a distributed database system including a plurality of nodes, the computer-readable medium including instructions for causing a computing system to perform the method of any one of claims 1 to 18.
20. An apparatus for managing transactions, the apparatus including:

a plurality of nodes arranged in a distributed database system, each node including at least one processor; and

a communication medium connecting ports of the plurality of nodes for sending and receiving information between the plurality of nodes;

wherein the plurality of nodes are configured to perform the method of any one of claims 1 to 18.