CLAIMS:

1. A method for performing a query operation on a database, the method comprising:

selecting a connection to the database from a plurality of connections to the database, the connections being arranged within a data structure and each having a predetermined number of query slots;

waiting for a query slot to be available on the selected connection;

in response to determining that the predetermined number of query slots comprise at least one available query slot, accessing one of the at least one available query slots;

reordering the plurality of connections within the data structure;

causing the query operation to execute at least partially using the accessed query slot; and

releasing the accessed query slot.

2. The method of claim 1, wherein causing the query operation to execute comprises asynchronously performing the query operation via the connection by sending the query operation.

3. The method of claim 2, further comprising receiving a query result.

4. The method of any one of claims 1 to 3, wherein the database is a sharded database and the method is performed for each shard.

5. The method of any one of claims 1 to 4, wherein the database is a multi-tenant database, and wherein the database remains available to access by each tenant while the method is being performed.

6. The method of any one of claims 1 to 5, wherein the data structure is at least one of a list, queue or heap.

7. The method of any one of the preceding claims, wherein the data structure is a priority heap and wherein selecting the connection to the database comprises selecting the connection at a root of the priority heap, and wherein reordering the plurality of connections within the data structure comprises placing the selected connection at a bottom of the priority heap to cause the priority heap to be re-ordered.

8. The method of claim 7, wherein the reordering is performed based on a <0, 0, >0 pattern of each connection's number of available query slots.

9. The method of any one of claims 1 to 8, further comprising:

waiting for access to a query runner module to become available before selecting the connection to the database,

in response to access to the query runner module being available, preventing other processes from accessing the query runner module; and

after reordering the data structure, allowing other processes to access the query runner module.

10. The method of claim 9, further comprising determining whether access to the query runner module is available by accessing query runner access control data.

11. The method of claim 10, wherein the query runner access control data includes at least one query runner access control value, and wherein preventing other processes from accessing the query runner module comprises setting the query runner access control value to a first value, and allowing other processes to access the query runner module comprises setting the query runner access control value to a second value.

12. The method of claim 11, wherein the query runner access control value comprises a query runner semaphore, so that preventing other processes from accessing the query runner

module comprises taking the query runner semaphore, and allowing other processes to access the query runner module comprises releasing the query runner semaphore.

13. The method of any one of claims 1 to 12 further comprising, after reordering the data structure, opening the selected connection.

14. The method of any one of claims 1 to 13, further comprising:

waiting for access to the connection to become available before performing the query operation,

in response to access to the connection being available, preventing other processes from performing respective query operations via the connection; and

after performing the query operation, allowing other processes to access the connection to perform the respective query operations.

15. The method of claim 14, further comprising determining whether access to a connection module is available by accessing connection access control data.

16. The method of claim 15, wherein the connection access control data includes at least one connection access control value, and wherein preventing other processes from accessing the connection comprises setting the connection access control value to a first value, and allowing other processes to access the connection comprises setting the connection access control value to a second value.

17. The method of claim 16, wherein the connection access control value comprises a connection semaphore, so that preventing other processes from accessing the connection comprises taking the connection semaphore, and allowing other processes to access the connection comprises releasing the connection semaphore.

18. The method of any one of claims 1 to 18, wherein waiting for the at least one query slot to be available on the selected connection comprises reading a maximum queries control value, taking one of the at least one available slots comprises one of incrementing or decrementing

the maximum queries control value, and releasing the accessed query slot comprises the other of incrementing or decrementing the maximum queries control value.

19. A device comprising:

a memory storing executable code;

a processor configured to access the memory to execute the executable code;

wherein when executing the executable code, the processor is caused to perform the method of any one of claims 1 to 18.

20. A computer-readable storage medium storing instructions that, when executed by a processor, cause the processor to perform the method of any one of claims 1 to 18.