CLAIMS

1. A middleware computer system for controlling transaction requests between applications and servers, the middleware system comprising:

a broker component configured to, at runtime:

define one or more collision groups of transaction requests based on previous or current observed transaction collisions; and

control delivery of transaction requests in parallel based on the one or more collision groups,

wherein the broker component comprises a receiver component configured to receive information about at least one of:

observed transaction collisions; and

further collision groups of transaction requests from a remote component, and wherein the broker component is configured to, at runtime, define one or more collision groups of transaction requests further based on the received information.

2. The middleware computer system of claim 1, wherein each of the one or more collision groups defines a plurality of transaction requests that if delivered in parallel will result in a transaction collision.

3. The middleware computer system of claim 1, wherein the broker component comprises a publisher component configured to communicate information about the one or more collision groups to one or more remote components.

~~4. The middleware computer system of claim 1, wherein the broker component comprises a receiver component configured to receive information about at least one of:~~

~~observed transaction collisions; and~~

~~further collision groups of transaction requests from a remote component, and wherein the broker component is configured to, at runtime, define one or more collision groups of transaction requests further based on the received information.~~

~~5.~~ 4. The middleware computer system of claim 1, wherein the broker component is configured to, responsive to identifying a plurality of transaction requests that are members of the same collision group of the one or more collision groups, queue the plurality of transaction requests to prevent parallel delivery of the plurality of transaction requests.

~~6.~~ 5. The middleware computer system of claim ~~5~~4, wherein the broker component is configured to queue the plurality of transaction requests in a first-in first-out order.

~~7.~~ 6. The middleware computer system of claim 1, ~~wherein~~Wherein the broker component is configured to, responsive to identifying a transaction collision resulting from delivery of a transaction request, redefine the one or more collision groups of transaction requests based on the identified transaction collision and control redelivery of the transaction request based on the redefined one or more collision groups.

~~8.~~ 7. The middleware computer system of claim 1, wherein the middleware system is configured to intercept and control delivery of transaction requests between a client application and a Multi-Version Concurrency Control, MVCC, constrained server.

~~9.~~ 8. A processor-implemented method for controlling transaction requests between applications and servers, the method comprising:

~~—~~defining, at runtime, one or more collision groups of transaction requests based on previous or current observed transaction collisions; ~~and~~

~~—~~controlling delivery of transaction requests in parallel based on the one or more collision groups~~.~~; and

receiving information about at least one of: observed transaction collisions; and further collision groups of transaction requests from a remote component,

wherein defining one or more collision groups of transaction requests is further based on the received information.~~10.~~

9. The method of claim ~~9~~8, wherein each of the one or more collision groups defines a plurality of transaction requests that if delivered in parallel will result in a transaction collision.

~~11.~~ 10. The method of claim ~~9~~8, further comprising:

~~—~~communicating information about the one or more collision groups to one or more remote components.

~~12.~~ 11. The method of claim ~~9~~8, further comprising:

~~receiving information about at least one of: observed transaction collisions; and further collision groups of transaction requests from a remote component,~~

~~and wherein defining one or more collision groups of transaction requests is further based on the received information.~~

~~13. The method of claim 9, further comprising:~~

responsive to identifying a plurality of transaction requests that are members of the same collision group of the one or more collision groups, queuing the plurality of transaction requests to prevent parallel delivery of the plurality of transaction requests.

12. The method of claim 11, wherein queuing the plurality of transaction requests comprises queuing the plurality of transaction requests in a first-in first-out order.

13. The method of claim 8, further comprising:

responsive to identifying a transaction collision resulting from delivery of a transaction request, redefining the one or more collision groups of transaction requests based on the identified, transaction collision; and

controlling redelivery of the transaction request based on the redefined one or more collision groups.

14. The method of claim 8, wherein the method is configured to intercept and control delivery of transaction requests between a client application and a Multi-Version Concurrency Control (MVCC) constrained server.

15. A computer program product comprising a computer-readable tangible storage medium having program instructions embodied therewith, the program instructions executable by a processing unit to cause the processing unit to perform, when run on a computer network, a method for controlling transaction requests between applications and servers, wherein the method comprises the steps of:

at runtime: defining, by a broker component, one or more collision groups of transaction requests based on previous or current observed transaction collisions; and

controlling delivery of transaction requests in parallel based on the one or more collision groups.

wherein the broker component comprises a receiver component configured to receive information about at least one of:

observed transaction collisions; and

further collision groups of transaction requests from a remote component, and wherein the broker component is configured to, at runtime, define one or more collision groups of transaction requests further based on the received information.

16. A computer system comprising at least one processor and the computer program product of claim 15, wherein the at least one processor is adapted to execute the computer program code of said computer program product.

17. ~~19.~~ The computer program product of claim ~~17~~15, wherein each of the one or more collision groups defines a plurality of transaction requests that if delivered in parallel will result in a transaction collision.

~~20.~~ 18. The computer program product of claim ~~17~~15, further comprising:

~~communicating~~ information about the one or more collision groups to one or more remote components.