CLAIMS

WHAT IS CLAIMED IS:

1. A computer-implemented method for managing a distributed memory caching system, comprising:

~~under the control of one or more computer systems configured with executable instructions,~~

monitoring, by a set of memory caching nodes in a cache cluster, for configuration changes made by other caching nodes in the~~a~~ set of ~~one or more~~ memory caching nodes ~~forming a cache cluster~~, each memory caching node of the set of ~~one or more~~ memory caching nodes providing storage for a portion of data contained in the cache cluster and ~~at least one~~each of the set of memory caching node~~s~~ of the cache cluster comprising a configuration for accessing the data from the set of ~~one or more~~ memory caching nodes, the configuration stored in a partitioned memory space of the set of memory caching nodes;

detecting, by a memory caching node of the set of memory caching nodes,~~determining~~ that a new memory caching node ~~be~~ has been added to the cache cluster, the detection that the new memory caching node has been added to the cache cluster being based at least in part on the monitoring by the set of memory caching nodes for configuration changes made by the other caching nodes;

provisioning the new memory caching node into the cache cluster; and

updating the configuration within the partitioned memory space of ~~the~~ at least one of the set of memory caching node~~s~~ based at least in part on the new memory caching node.

2. The computer-implemented method of claim 1:

wherein the monitoring of the set of ~~one or more~~ memory caching nodes further comprises detecting a failure of a failed memory caching node from the cache cluster;

wherein provisioning the new memory caching further comprises causing a replacement memory caching node to be provisioned in place of the failed memory caching node; and

wherein updating the configuration within the at least one of the set of memory caching node~~s~~ further comprises updating the configuration within the partitioned memory space of the at least one of the set of memory caching nodes based at least in part on the replacement of the failed memory caching node with the replacement memory caching node.

3. The computer-implemented method of claim 1:

wherein ~~the~~ monitoring the set of ~~one or more~~ memory caching nodes further comprises detecting an indicator indicating a need of additional caching performance from the cache cluster;

wherein provisioning the new memory caching node into the cache cluster further comprises provisioning the new memory caching node in place of a replaced memory caching node that has less resources than the new memory caching node; and

wherein updating the configuration within the partitioned memory space of the at least one of the memory caching nodes further comprises updating the configuration within the at least one of the set of memory caching nodes based at least in part on the replacement of the replaced memory caching node with the new memory caching node.

4. The computer-implemented method of claim 1, further comprising updating an alias of a configuration endpoint based at least in part on the new memory caching node.

5. The computer-implemented method of claim 1, wherein provisioning the new memory caching node into the cache cluster further comprises constructing a workflow that includes the operations of:

provisioning the new memory caching node;

adding the memory caching node to the set of one or more memory caching nodes;

determining a new configuration; and

updating the configuration within the partitioned memory space of each of the set of memory caching nodes.

6. The method of claim 1, wherein information in the partitioned memory space of the set of memory caching nodes is accessed according to an extension of a standard caching protocol, the standard caching protocol comprising at least one of an eviction procedure or an expiration procedure.

7. The method of claim 6, wherein the partitioned memory space is excluded from being subject to the eviction procedure and the expiration procedure defined by the standard caching protocol.

~~5~~8. A computer-implemented method for managing a service, comprising:

~~under the control of one or more computer systems configured with executable instructions,~~

monitoring, by a set of storage nodes in a cluster, for configuration changes of the cluster made by other storage nodes of the set of storage nodes, ~~a~~ the cluster comprising ~~one or~~

~~more~~the set of storage nodes, each storage node of the ~~one or more~~set of storage nodes providing storage for a portion of data contained in a distributed data store;

enabling storage in ~~at least one~~each of the set of storage node~~s~~, of ~~storing~~ a configuration for accessing the data within the cluster, the configuration stored in a partitioned memory space of the set of storage nodes;

detecting that the configuration has been updated, the detection being based at least in part on the monitoring by the set of storage nodes for configuration changes of the cluster made by the other storage nodes; and

as a result of ~~a~~ the determination that the configuration be updated, updating the configuration within the partitioned memory space of ~~the~~ at least one of the set of storage node~~s~~.

~~6~~9. The computer-implemented method of claim ~~5~~8, wherein updating the configuration further comprises updating the configuration based at least in part on a modification to the cluster.

~~7~~10. The computer-implemented method of claim ~~6~~9, further comprising modifying the cluster by removing a storage node from the cluster through the operations of:

de-provisioning the storage node; and

reallocating at least a portion of data contained in the distributed data store among at least a plurality of remaining storage nodes in the cluster.

~~8~~11. The computer-implemented method of claim ~~6~~9, further comprising modifying the cluster by adding a new storage node to the cluster through the operations of:

provisioning the new storage node; and

reallocating at least a portion of data contained in the distributed data store among at least a plurality of storage nodes in the cluster.

~~9~~12. The computer-implemented method of claim ~~6~~9, further comprising modifying the cluster by replacing a first storage node in the cluster through the operations of:

provisioning a new storage node;

de-provisioning the first storage node; and

allocating at least a portion of data from the first storage node to the new storage node.

13. The computer-implemented method of claim 9, further comprising modifying the cluster by altering a characteristic of a storage node from the cluster.

14. The computer-implemented method of claim 13, wherein altering the characteristic of the storage node further comprises modifying a storage size of the storage node.

~~10~~15. A computer system for managing a caching service, comprising:

one or more computing resources having one or more processors and memory including executable instructions that, when executed by the one or more processors, cause the one or more processors to implement at least:

~~one or more memory~~a set of memory caching nodes forming a cache cluster that:

provide storage for a distributed data cache allocated across the cache cluster; ~~and~~

provide storage ~~for~~ of a configuration for accessing data within the cache cluster, the configuration stored in a partitioned memory space of the set of memory caching nodes; ~~and~~

monitor for configuration changes made by other caching nodes in the cluster;

detect changes to the configuration of the cache cluster based at least in part on the monitoring of the set of memory caching nodes; and

update the configuration within the partitioned memory space of at least one of the set of memory caching nodes based at least in part on detecting the changes to the configuration of the cache cluster; and

a management system that monitors and provisions memory caching nodes and updates the configuration within the partitioned memory space of at least one of the set of memory caching nodes, the updated configuration of the cache cluster based at least in part on ~~changes~~ alterations to the cache cluster caused by the management system.

~~11~~16. The computer system of claim ~~10~~15, further comprising a data store that provides data for the distributed data cache that is allocated across the cache cluster.

~~12~~17. The computer system of claim ~~11~~16, wherein the data store is a relational database and wherein the distributed data cache includes responses to queries from the relational database.

~~13~~18. The computer system of claim ~~10~~15, further comprising a configuration endpoint that provides an alias to a memory caching node from the cache cluster and receives alias updates from the management system.

~~14. The computer system of claim 13, wherein the configuration endpoint has a static address at which it can be used to resolve the alias to the memory caching node.~~

~~15. The computer system of claim 10, further comprising a program execution environment in which the cache cluster and management system are allocated computing resources on which to execute.~~

19. The computer system of claim 18, wherein the configuration endpoint has a static address, that is resolvable to a particular memory caching node of the cache cluster, wherein resolving the alias to the particular memory caching node causes the particular memory caching node to provide an initial configuration of the cluster to a client device, and wherein the static address of the particular memory caching node remains constant for a period of time.

20. The computer system of claim 15, further comprising a program execution environment in which the cache cluster and management system are allocated computing resources on which to execute.

21. One or more computer-readable storage media having collectively stored thereon executable instructions that, when executed by one or more processors of a computer system, cause the computer system to at least:

monitor, by a set of memory caching nodes of a cache cluster, the cache cluster for indications of failure of one or more of the set of memory caching nodes, the set of memory caching nodes providing storage for a portion of data contained in a distributed data cache;

determine, by a memory caching node of the set of memory caching nodes, that an indicator of a particular memory caching node from the cache cluster has passed a failure threshold, the determination that the indicator has passed the failure threshold being based at least in part on the monitoring by the memory caching node of the set of memory caching nodes;

cause, by the memory caching node, the particular memory caching node to be removed from the cache cluster; and

update, by the memory caching node, a configuration stored in a partitioned memory space of each of the set of memory caching nodes of the cache cluster based at least in part on the determination that the indicator of the particular memory caching node has passed the failure threshold.

22. The computer-readable storage media of claim 21, wherein the instructions further comprise instructions that, when executed, cause the computer system to at least:

cause a new memory caching node to be provisioned and added to the cache cluster; and

update, by the memory caching node, the configuration within at least one memory caching node from the cache cluster based at least in part on the provisioned new memory caching node.

23. The computer-readable storage media of claim 22, wherein updating the configuration within the at least one memory caching node from the cache cluster further comprises updating the configuration such that the new node replaces the removed memory caching node.

24. The computer-readable storage media of claim 21, wherein the configuration comprises connection information to the cache cluster.

25. The computer-readable storage media of claim 21, wherein updating the configuration within the at least one memory caching node from the cache cluster further comprises updating the memory caching nodes with the configuration to cause the memory caching nodes to propagate the configuration using peer-to-peer updating.

26. The computer-readable storage media of claim 25, wherein causing the memory caching nodes to propagate the configuration further comprises updating a configuration endpoint with a new alias to a memory caching endpoint within the set of one or more memory caching endpoints.

27. The computer-readable storage media of claim 21, wherein updating the configuration stored by the at least one memory caching node further comprises receiving an indication from a management system of one or more assignments of memory caching nodes for use with a consistent hashing method.

CLAIMS

WHAT IS CLAIMED IS:

1. A computer-implemented method for managing a distributed memory caching system, comprising:

monitoring, by a set of memory caching nodes in a cache cluster, for configuration changes made by other caching nodes in the set of memory caching nodes, each memory caching node of the set of memory caching nodes providing storage for a portion of data contained in the cache cluster and each of the set of memory caching nodes of the cache cluster comprising a configuration for accessing the data from the set of memory caching nodes, the configuration stored in a partitioned memory space of the set of memory caching nodes;

detecting, by a memory caching node of the set of memory caching nodes, that a new memory caching node has been added to the cache cluster, the detection that the new memory caching node has been added to the cache cluster being based at least in part on the monitoring by the set of memory caching nodes for configuration changes made by the other caching nodes;

provisioning the new memory caching node into the cache cluster; and

updating the configuration within the partitioned memory space of at least one of the set of memory caching nodes based at least in part on the new memory caching node.

2. The computer-implemented method of claim 1:

wherein the monitoring of the set of memory caching nodes further comprises detecting a failure of a failed memory caching node from the cache cluster;

wherein provisioning the new memory caching further comprises causing a replacement memory caching node to be provisioned in place of the failed memory caching node; and

wherein updating the configuration within the at least one of the set of memory caching nodes further comprises updating the configuration within the partitioned memory space of the at least one of the set of memory caching nodes based at least in part on the replacement of the failed memory caching node with the replacement memory caching node.

3. The computer-implemented method of claim 1:

wherein monitoring the set of memory caching nodes further comprises detecting an indicator indicating a need of additional caching performance from the cache cluster;

wherein provisioning the new memory caching node into the cache cluster further comprises provisioning the new memory caching node in place of a replaced memory caching node that has less resources than the new memory caching node; and

wherein updating the configuration within the partitioned memory space of the at least one of the memory caching nodes further comprises updating the configuration within the at least one of the set of memory caching nodes based at least in part on the replacement of the replaced memory caching node with the new memory caching node.

4. The computer-implemented method of claim 1, further comprising updating an alias of a configuration endpoint based at least in part on the new memory caching node.

5. The computer-implemented method of claim 1, wherein provisioning the new memory caching node into the cache cluster further comprises constructing a workflow that includes the operations of:

provisioning the new memory caching node;

adding the memory caching node to the set of one or more memory caching nodes;

determining a new configuration; and

updating the configuration within the partitioned memory space of each of the set of memory caching nodes.

6. The method of claim 1, wherein information in the partitioned memory space of the set of memory caching nodes is accessed according to an extension of a standard caching protocol, the standard caching protocol comprising at least one of an eviction procedure or an expiration procedure.

7. The method of claim 6, wherein the partitioned memory space is excluded from being subject to the eviction procedure and the expiration procedure defined by the standard caching protocol.

8. A computer-implemented method for managing a service, comprising:

monitoring, by a set of storage nodes in a cluster, for configuration changes of the cluster made by other storage nodes of the set of storage nodes, the cluster comprising the set of

storage nodes, each storage node of the set of storage nodes providing storage for a portion of data contained in a distributed data store;

enabling storage in each of the set of storage nodes, of a configuration for accessing the data within the cluster, the configuration stored in a partitioned memory space of the set of storage nodes;

detecting that the configuration has been updated, the detection being based at least in part on the monitoring by the set of storage nodes for configuration changes of the cluster made by the other storage nodes; and

as a result of the determination that the configuration be updated, updating the configuration within the partitioned memory space of at least one of the set of storage nodes.

9. The computer-implemented method of claim 8, wherein updating the configuration further comprises updating the configuration based at least in part on a modification to the cluster.

10. The computer-implemented method of claim 9, further comprising modifying the cluster by removing a storage node from the cluster through the operations of:

de-provisioning the storage node; and

reallocating at least a portion of data contained in the distributed data store among at least a plurality of remaining storage nodes in the cluster.

11. The computer-implemented method of claim 9, further comprising modifying the cluster by adding a new storage node to the cluster through the operations of:

provisioning the new storage node; and

reallocating at least a portion of data contained in the distributed data store among at least a plurality of storage nodes in the cluster.

12. The computer-implemented method of claim 9, further comprising modifying the cluster by replacing a first storage node in the cluster through the operations of:

provisioning a new storage node;

de-provisioning the first storage node; and

allocating at least a portion of data from the first storage node to the new storage node.

13. The computer-implemented method of claim 9, further comprising modifying the cluster by altering a characteristic of a storage node from the cluster.

14. The computer-implemented method of claim 13, wherein altering the characteristic of the storage node further comprises modifying a storage size of the storage node.

15. A computer system for managing a caching service, comprising:

one or more computing resources having one or more processors and memory including executable instructions that, when executed by the one or more processors, cause the one or more processors to implement at least:

a set of memory caching nodes forming a cache cluster that:

provide storage for a distributed data cache allocated across the cache cluster;

provide storage of a configuration for accessing data within the cache cluster, the configuration stored in a partitioned memory space of the set of memory caching nodes;

monitor for configuration changes made by other caching nodes in the cluster;

detect changes to the configuration of the cache cluster based at least in part on the monitoring of the set of memory caching nodes; and

update the configuration within the partitioned memory space of at least one of the set of memory caching nodes based at least in part on detecting the changes to the configuration of the cache cluster; and

a management system that monitors and provisions memory caching nodes and updates the configuration within the partitioned memory space of at least one of the set of memory caching nodes, the updated configuration of the cache cluster based at least in part on alterations to the cache cluster caused by the management system.

16. The computer system of claim 15, further comprising a data store that provides data for the distributed data cache that is allocated across the cache cluster.

17. The computer system of claim 16, wherein the data store is a relational database and wherein the distributed data cache includes responses to queries from the relational database.

18. The computer system of claim 15, further comprising a configuration endpoint that provides an alias to a memory caching node from the cache cluster and receives alias updates from the management system.

19. The computer system of claim 18, wherein the configuration endpoint has a static address, that is resolvable to a particular memory caching node of the cache cluster, wherein resolving the alias to the particular memory caching node causes the particular memory caching node to provide an initial configuration of the cluster to a client device, and wherein the static address of the particular memory caching node remains constant for a period of time.

20. The computer system of claim 15, further comprising a program execution environment in which the cache cluster and management system are allocated computing resources on which to execute.

21. One or more computer-readable storage media having collectively stored thereon executable instructions that, when executed by one or more processors of a computer system, cause the computer system to at least:

monitor, by a set of memory caching nodes of a cache cluster, the cache cluster for indications of failure of one or more of the set of memory caching nodes, the set of memory caching nodes providing storage for a portion of data contained in a distributed data cache;

determine, by a memory caching node of the set of memory caching nodes, that an indicator of a particular memory caching node from the cache cluster has passed a failure threshold, the determination that the indicator has passed the failure threshold being based at least in part on the monitoring by the memory caching node of the set of memory caching nodes;

cause, by the memory caching node, the particular memory caching node to be removed from the cache cluster; and

update, by the memory caching node, a configuration stored in a partitioned memory space of each of the set of memory caching nodes of the cache cluster based at least in part on the determination that the indicator of the particular memory caching node has passed the failure threshold.

22. The computer-readable storage media of claim 21, wherein the instructions further comprise instructions that, when executed, cause the computer system to at least:

cause a new memory caching node to be provisioned and added to the cache cluster; and

update, by the memory caching node, the configuration within at least one memory caching node from the cache cluster based at least in part on the provisioned new memory caching node.

23. The computer-readable storage media of claim 22, wherein updating the configuration within the at least one memory caching node from the cache cluster further comprises updating the configuration such that the new node replaces the removed memory caching node.

24. The computer-readable storage media of claim 21, wherein the configuration comprises connection information to the cache cluster.

25. The computer-readable storage media of claim 21, wherein updating the configuration within the at least one memory caching node from the cache cluster further comprises updating the memory caching nodes with the configuration to cause the memory caching nodes to propagate the configuration using peer-to-peer updating.

26. The computer-readable storage media of claim 25, wherein causing the memory caching nodes to propagate the configuration further comprises updating a configuration endpoint with a new alias to a memory caching endpoint within the set of one or more memory caching endpoints.

27. The computer-readable storage media of claim 21, wherein updating the configuration stored by the at least one memory caching node further comprises receiving an indication from a management system of one or more assignments of memory caching nodes for use with a consistent hashing method.