Edit Active-Active database configuration

An Active-Active database (formerly known as CRDB) is a database that spans multiple Redis Enterprise clusters. The clusters that host instances of the Active-Active database are called participating clusters. When you create an Active-Active database you must specify the participating clusters that host the Active-Active database instances.

When you edit the database configurations of an Active-Active database, most database settings only apply to the Active-Active database instance that you are editing, including:

- Memory limit
- Data persistence
- Redis password
- Number of shards
- TLS mode
- Periodic backup

You can change the global configuration of the Active-Active database from the command-line with the `crdb-cli`.

Participating clusters

You can add and remove participating clusters of an Active-Active database to change the Active-Active topology. Use the participating clusters list to manage the changes to Active-Active topology in the UI. You can make multiple changes at once to the list of participating clusters. The changes you make to the list are committed when the database configuration is saved.

All of the existing participating clusters must be online and in a syncing state when you add new participating clusters.

After you add new participating clusters to an existing Active-Active database, the new Active-Active database instance can accept connections and read operations. The new instance does not accept write operations until it is in the syncing state. New participating clusters create the Active-Active database instance based on the global Active-Active database configuration.

All of the existing participating clusters must be online and in a syncing state when you remove an online participating clusters. If you must remove offline participating clusters, you can do this with forced removal. If a participating cluster that was removed forcefully returns attempts to re-join the cluster, it will have an out of date on Active-Active database membership. The joined participating clusters reject updates sent from the removed participating cluster.

Database replication backlog
Redis databases that use replication for high availability maintain a replication backlog (per shard) to synchronize the primary and replica shards of a database. By default, the replication backlog is set to one percent (1%) of the database size divided by the database number of shards and ranges between 1MB to 250MB per shard. Use the `rladmin` and the `crdb-cli` utilities to control the size of the replication backlog. You can set it to auto or set a specific size.

The syntax varies between regular and Active-Active databases.

For a regular Redis database:

```
rladmin tune db <db:id | name> repl_backlog <Backlog size in MB or 'auto'>
```

For an Active-Active database:

```
crdb-cli crdb update --crdb-guid <crdb_guid> --default-db-config "{\"repl_backlog_size\": <size in MB | 'auto'>}"
```

### Active-Active CRDT replication backlog

In addition to the database replication backlog, Active-Active databases maintain a CRDT replication backlog (per shard) to synchronize the database instances between clusters. By default, the CRDT replication backlog is set to one percent (1%) of the Active-Active database size divided by the database number of shards and ranges between 1MB to 250MB per shard. Use the `crdb-cli` utility to control the size of the CRDT replication backlog. You can set it to auto or set a specific size:

```
crdb-cli crdb update --crdb-guid <crdb_guid> --default-db-config "{\"crdt_repl_backlog_size\": <size in MB | 'auto'>}"
```

### Active-Passive replication backlog

In addition to the database replication backlog, Active-Passive databases maintain a replication backlog (per shard) to synchronize the database instances between clusters. By default, the replication backlog is set to one percent (1%) of the database size divided by the database number of shards and ranges between 1MB to 250MB per shard. Use the `rladmin` utility to control the size of the replication backlog. You can set it to auto or set a specific size.

For an Active-Passive database:

```
rladmin tune db <db:id | name> repl_backlog <Backlog size in MB or 'auto'>
```

Note: On an Active-Passive database, the replication backlog configuration applies to both the replication backlog for shards synchronization and for synchronization of database instances between clusters.

For Redis Software versions earlier than 6.0.20: The replication backlog and the CRDT replication backlog defaults are set to 1MB and cannot be set dynamically with 'auto' mode. To control the size of the replication log, use `rladmin` to tune the local database instance in each cluster.

```
rladmin tune db <db:id | name> repl_backlog <Backlog size in MB (or if ending with bytes, KB or GB, in the respective unit)>
```