Active-Active Geo-Distributed Redis

In Redis Enterprise, active-active geo-distribution is based on CRDT technology. The Redis Enterprise implementation of CRDT is called an Active-Active database (formerly known as CRDB). With Active-Active databases, applications can read and write to the same data set from different geographical locations seamlessly and with latency less than one millisecond (ms), without changing the way the application connects to the database.

Active-Active databases also provide disaster recovery and accelerated data read-access for geographically distributed users.

Note: Active-Active databases do not replicate the entire database, only the data. Database configurations, LUA scripts, and other support info are not replicated.

You can create Active-Active databases on Redis Enterprise Software or Redis Cloud.

Considerations for Active-Active Databases

Active-Active databases are based on multi-master replication that is configured to run on each database. An Active-Active database is made up of instances of the data that are each stored on an RS cluster.

Before configuring an Active-Active database, you must:

- If the Active-Active database spans a WAN, establish a VPN between each network that hosts a cluster with an instance.
- Setup RS clusters for each Active-Active database instance.
  
  All clusters must have the same Redis Enterprise Software version.
- Configure FQDNs in a DNS server for connections to the cluster.
  
  Active-Active databases are not compatible with the Discovery Service for inter-cluster communications, but are compatible with local application connections.
- Configure the network so that all nodes in each cluster can connect to the proxy port and the cluster admin port (9443) of each cluster.
- Confirm that a network time service is configured and running on each node in all clusters.

Redis Modules on Active-Active Databases

Active-Active databases support only compatible Redis modules.

- RediSearch 2.x in Redis Enterprise Software (RS) 6.0 and higher.
- RedisGears

Active-Active database current limitations

1. The RS admin console is limited to five participating clusters or instances in an Active-Active database.

2. An existing database cannot be changed into an Active-Active database. To move data from an existing database to an Active-Active database you must create a new Active-Active database and migrate the data.

3. Active-Active databases require FQDNs or mDNS (development only). Discovery Service is not supported with Active-Active databases.

4. Active-Active databases are not compatible with Replica Of.
Network Time Service (NTP or Chrony)

For Active-Active databases, you must use a time service like NTP or Chrony. This is critical to minimize time drift both intercluster and intracluster for Active-Active databases on an ongoing basis.

There may be times that the OS system time is used for conflict resolution between instances of an Active-Active database, although that rarely happens. The built-in vector clocks tell RS the order of operations, or identifies that the data operations were concurrent. When there is no option to intelligently handle conflicting writes, OS timestamps are used in resolving the conflict. For example, in certain cases "string type" uses timestamps to resolve conflicts.

The RS installation checks if there is a network time service installed, running, and configured to start on boot.

- If no network time service is found, the installation asks if you want to “tune the system”.
- If you answer yes, you are prompted to install and configure a network time service.
- If you answer yes, the NTP is installed.

For example:

```
2017-10-30 11:24:07 [?] Do you want to automatically tune the system for best performance [Y/N]? Y
2017-10-30 11:24:15 [?] Cluster nodes must have their system time synchronized.
Do you want to set up NTP time synchronization now [Y/N]? Y
2017-10-30 11:24:19 [.] Making sure NTP is installed and time is set.
```

Network configurations

RS assumes that networking between the clusters is already configured when you create an Active-Active database. For security purposes, we recommend that you configure a secure VPN between all clusters that host an instance of an Active-Active database. The setup of the Active-Active database fails if there is no connectivity between the clusters.

Network ports

For initial configuration and ongoing maintenance of an Active-Active database, every node must have access to the REST API ports of every other node. You must also open ports for VPNs and Security groups.

For synchronization, Active-Active databases operate over the standard endpoint ports. The endpoint port that you configure when you create the Active-Active database is the endpoint port of the proxy for that Active-Active database on each cluster.

Data persistence

You can set the data persistence configuration, including AOF (Append-Only File) data persistence and snapshot, for each participating cluster.

Syncer process

Each node in a cluster containing an instance of an Active-Active database hosts a process called syncer. The syncer process:

1. Connects to the other cluster proxy
2. Reads data from that database
3. Writes the data to the master shard of that database

Some replication capabilities are also included in open source redis.

The Master at the top of the master-replica tree creates a replication ID. This replication ID is identical for all replicas in that tree. When a new master is appointed, the replication ID changes but a partial sync from the previous ID is still possible. In a partial sync, the
backlog of operations since the offset are transferred as raw operations. In a full sync, the data from the master is transferred to the replica as an RDB file which is followed by a partial sync.

Partial synchronization requires a backlog large enough to store the data operations until connection is restored.

**Syncer in Active-Active Replication**

In the case of an Active-Active database:

- Multiple past replication IDs and offsets are stored to allow for multiple syncs
- The Active-Active backlog is also sent to the replica during a full sync

---

**Warning** - Full sync triggers heavy data transfers between geo-replicated instances of an Active-Active database.

---

The scenarios in which an Active-Active database updates to other instances use partial synchronization are:

- Failover of master shard to replica shard
- Restart or crash of replica shard that requires sync from master
- Migrate replica shard to another node
- Migrate master shard to another node as a replica using failover and replica migration
- Migrate master shard and preserve roles using failover, replica migration, and second failover to return shard to master

---

**Note:** Synchronization of data from the master shard to the replica shard is always a full synchronization.

---

**Updated:** September 23, 2021