### Configuring Shard Placement

In Redis Enterprise Software, the location of master and slave shards on the cluster nodes can impact the database and node performance. Master shards and their corresponding slave shards are always placed on separate nodes for data resiliency. The **shard placement policy** helps to maintain optimal performance and resiliency.

In addition to the shard placement policy, considerations that determine shard placement are:

- Separation of master and replica shards
- Available persistence and Redis on Flash (RoF) storage
- Rack awareness
- Memory available to host the database when fully populated

The shard placement policies are:

- **dense** - Place as many shards as possible on the smallest number of nodes to reduce the latency between the proxy and the database shards; Recommended for Redis on RAM databases to optimize memory resources
- **sparse** - Spread the shards across as many nodes in the cluster as possible to spread the traffic across cluster nodes; Recommended for Redis on Flash databases to optimize disk resources

When you create a Redis Enterprise Software cluster, the default shard placement policy (**dense**) is assigned to all databases that you create on the cluster.

You can:

- Change the default shard placement policy for the cluster to **sparse** so that the cluster applies that policy to all databases that you create
- Change the shard placement policy for each database after the database is created

### Default shard placement policy

When you create a new cluster, the cluster configuration has a **dense** default shard placement policy. When you create a database, this default policy is applied to the new database.

To see the current default shard placement policy, run `rladmin info cluster`:

```bash
bash-4.2$ rladmin info cluster
Cluster configuration:
repl_disksless: enabled
shards_overbooked: disabled
default_non_sharded_proxy_policy: single
default_sharded_proxy_policy: single
default_shards_placement: sparse
```

To change the default shard placement policy so that new databases are created with the **sparse** shard placement policy, run:

```
rladmin tune cluster default_shards_placement [ dense | sparse ]
```

### Shard placement policy for a database
To see the shard placement policy for a database in `rladmin status`:

<table>
<thead>
<tr>
<th>DB-ID</th>
<th>NAME</th>
<th>TYPE</th>
<th>STATUS</th>
<th>SHARDS</th>
<th>PLACEMENT</th>
<th>REPLICAITON</th>
<th>PERSISTENCE</th>
<th>ENDPOINT</th>
</tr>
</thead>
<tbody>
<tr>
<td>db:1</td>
<td>db1</td>
<td>redis</td>
<td>active</td>
<td>2</td>
<td>sparse</td>
<td>disabled</td>
<td>disabled</td>
<td>redis-18051.cluster1.local:18051</td>
</tr>
<tr>
<td>db:2</td>
<td>db2</td>
<td>redis</td>
<td>active</td>
<td>2</td>
<td>dense</td>
<td>disabled</td>
<td>disabled</td>
<td>redis-13868.cluster1.local:13868</td>
</tr>
</tbody>
</table>

To change the shard placement policy for a database, run:

```
rladmin placement db [ database name | database ID ] [ dense | sparse ]
```

Updated: January 6, 2022