Redis on Flash

Redis on Flash (RoF) offers users of Redis Enterprise Software and Redis Enterprise Cloud the unique ability to have large Redis databases but at significant cost savings. Where standard Redis databases must all be in RAM, Redis on Flash enables your Redis databases to span both RAM and dedicated flash memory (SSD). Whilst keys are always stored in RAM, RoF intelligently manages the location of their values (RAM vs Flash) in the database via a LRU-based (least-recently-used) mechanism. Hot values are stored in RAM, but infrequently used, or warm values, are ejected to flash memory. This enables you to have much larger datasets with RAM-like latency and performance, but at dramatically lower cost than an all-RAM database.

By using Redis on Flash to distribute the data between RAM and flash memory, which is much cheaper than RAM, you can lower your TCO and better utilize hardware, hypervisor, and cloud resources. In many cases, Redis on Flash can cut resource costs by over 70% when compared to an all-RAM Redis Enterprise Software deployment.

Flash memory

Unlike standard Redis Enterprise Software installations, implementing Redis on Flash requires pre-planning around memory and overall sizing. There are a few critical recommendations:

- The flash memory should be local to the server/VM/instance/container as opposed to network attached.
- The flash memory should be dedicated to RoF and not shared with other parts of the database, (e.g. durability, binaries, etc.).
- The flash memory should be NVMe based for best performance.

For more information read Ephemeral and Persistent Storage in Redis Enterprise Software.

When running on a cloud environment, the flash memory for Redis on Flash should be on the ephemeral SSDs of the cloud instance and persistent database storage should be network attached, e.g AWS EBS. For AWS, we specifically recommend “Storage Optimized I3 - High I/O Instances” because of the performance of NVMe for flash memory.

When deploying RoF on an on-premises environment, local SSD (preferably NVMe-based) should be used. To be clear, running RoF over Network Attached Storage (NAS), Storage Area Network (SAN), or with local HDD drives isn't supported.

Please note - the Redis Enterprise Software database persistent and ephemeral storage should be on different disks, either local or attached.

When you begin planning the deployment of Redis on Flash in Production, we recommend working closely with the Redis technical team for sizing and performance tuning.
Tunable RAM to flash ratio

You can easily configure or tune the ratio of RAM-to-Flash for each database independently, optimizing performance for your specific use case. This is an online operation requiring no downtime for your database. Think of this like a gas pedal in a car, the database speeds up as you give it more gas (RAM). We recommend you keep at least 20% of all values in RAM.

Working set management

Of your dataset, perhaps there is a subset of highly active objects considered the application’s “working set.” Redis on Flash intelligently manages the location of the working set (RAM) and the infrequently accessed keys (flash memory), based on LRU (least-recently-used) on a per-object basis.

Redis client support

Just like all-RAM databases, RoF is compatible with existing Redis applications. Databases that employ RoF are identical to all-RAM Redis Enterprise Software databases in characteristics and features.

Redis on Flash vs disk-based databases

Flash memory can be SATA or NVMe based storage devices, but NVMe is where you can see the performance benefits. There are many databases in the industry that are disk based. Disk-based databases use RAM to cache part of the data for fast access. However, this approach is different than extending RAM in a number of ways.

- **Hot Value Handling**: In many applications, a large portion of operations utilize only a subset of keys in the database. For example, the same keys may receive multiple writes repeatedly in a short amount of time. Under these conditions, disk-based databases have to repeatedly do I/O to save the updates to disk. In contrast, with RoF, the hot values stay in RAM and repeated writes to the same key do not generate I/O to the flash memory.

- **Write Performance**: With pure disk-based databases, the writes to the disk have to be durable. To ensure durable writes, disk-based databases use techniques like WAL (write-ahead log) or Redo-Logs. In contrast, when RoF ejects a value from RAM to Flash, the write operation does not incur the expensive WAL or Redo log techniques. In other words, write amplification with durable writes is much slower than writes RoF performs to extend RAM. That said, you can still do durable writes with RoF, but there are some considerations.

- **Future Proof**: In recent years, with the emergence of persistent memory technologies, memory has been moving to converge with storage. Persisted memory technologies like 3D XPoint are good examples of these technologies. These technologies assume that your application is aware of which parts of the dataset should be kept in RAM and which part is OK to store in persistent memory. If your application is not specifically designed for this technology, persistent-memory performance is going to be very slow and perhaps unpredictable. Redis on Flash, in contrast, is application agnostic as it performs this function on the server side and your application has no need to understand where the data resides. Your application issues the same commands it always has with Redis and Redis Enterprise Software. You get the benefits of RoF now and into the future regardless of how flash memory evolves.

Next steps

To create Redis on Flash databases, you must meet the following prerequisites:

- Mount ephemeral and persistent storage on Redis Enterprise Software nodes with the proper disk size.
- Meet all hardware requirements and software requirements.
- Install Redis Enterprise Software.

Once these requirements are met, both Redis on Flash databases and all-RAM databases can be created and managed in the same cluster. For additional details, refer to Creating a new database.
When Redis on Flash is enabled, additional settings and metrics are available in the system.

Updated: April 5, 2022