

## Definitions

### AWS - ElastiCache

- **ElastiCache** improves the performance of your database through caching query results.
- The primary purpose of an in-memory key-value store is to provide ultra-fast (sub-millisecond latency) and inexpensive access to copies of data.
- Most data stores have areas of data that are frequently accessed but seldom updated.
- Additionally, querying a database is always slower and more expensive than locating a key in a key-value pair cache.
- Some database queries are especially expensive to perform, for example, queries that involve joins across multiple tables or queries with intensive calculations.
- By caching such query results, you pay the price of the query once and then are able to quickly retrieve the data multiple times without having to re-execute the query.

Database -> ElastiCache.

#### Cluster Engine

- Redis
  - Multi-AZ cluster
  - HIPAA or PIC-DSS compliance
- Memcached
  - High performance
  - Distributed

Redis and Memcached are popular, open-source, in-memory data stores.

Although they are both easy to use and offer high performance, there are important differences to consider when choosing an engine. Memcached is designed for simplicity while Redis offers a rich set of features that make it effective for a wide range of use cases.

In this scenario,

Redis can provide a much more durable and powerful cache layer to the prototype distributed system, however, you should take note of one keyword in the requirement: **multithreaded**.

In terms of commands execution, **Redis is mostly a single-threaded server.**

It is not designed to benefit from multiple CPU cores unlike Memcached, however, you can launch several Redis instances to scale out on several cores if needed.

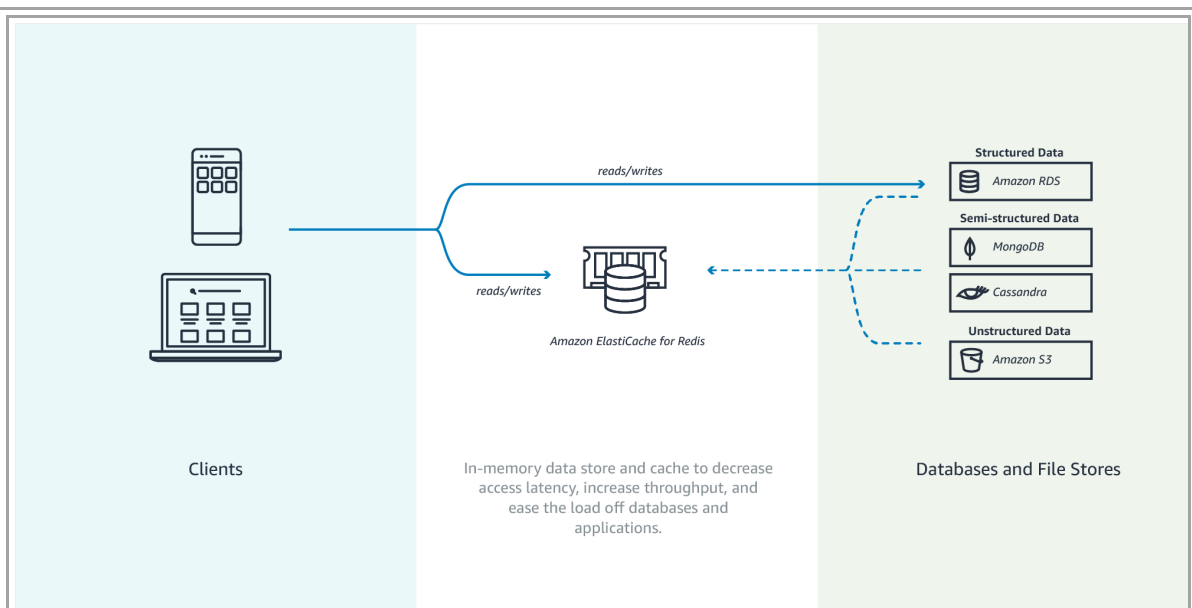
Memcached is a more suitable choice since the scenario specifies that the system will run large nodes with multiple cores or threads and in addition, the prototype only needs simple data structures which Memcached can adequately provide.

Moreover, the ability to enable the client programs to automatically identify all of the nodes in a cache cluster can be met by using the Auto Discovery feature of Amazon ElastiCache for Memcached. This is why the most suitable answer for this scenario is ***Amazon ElastiCache for Memcached***.

**You can choose Memcached over Redis if you have the following requirements:**

- You need the simplest model possible.
- You need to run large nodes with multiple cores or threads.
- You need the ability to scale out and in, adding and removing nodes as demand on your system increases and decreases.
- You need to cache objects, such as a database.

	Memcached	Redis
Sub-millisecond latency	Yes	Yes
Developer ease of use	Yes	Yes
Data partitioning	Yes	Yes
Support for a broad set of programming languages	Yes	Yes
Advanced data structures	-	Yes
Multithreaded architecture	Yes	-
Snapshots	-	Yes
Replication	-	Yes
Transactions	-	Yes
Pub/Sub	-	Yes
Lua scripting	-	Yes
Geospatial support	-	Yes



### Create your Amazon ElastiCache cluster

Cluster engine ☐ Redis  
In-memory data structure store used as database, cache and message broker. ElastiCache for Redis offers Multi-AZ with Auto-Failover and enhanced robustness.

Memcached settings

Engine version

Parameters

Number of nodes

Advanced Memcached settings

#### Select node type

Instance family	r5	m5	r4	m4	r3	m3	t2
Node type	cache.r5.large	cache.r5.xlarge	cache.r5.2xlarge	cache.r5.4xlarge	cache.r5.12xlarge	cache.r5.24xlarge	
Memory (GiB)	13.07	26.32	52.82	105.81	317.77	635.61	
Network performance	Up to 10 Gigabit	Up to 10 Gigabit	Up to 10 Gigabit	Up to 10 Gigabit	10 Gigabit	25 Gigabit	

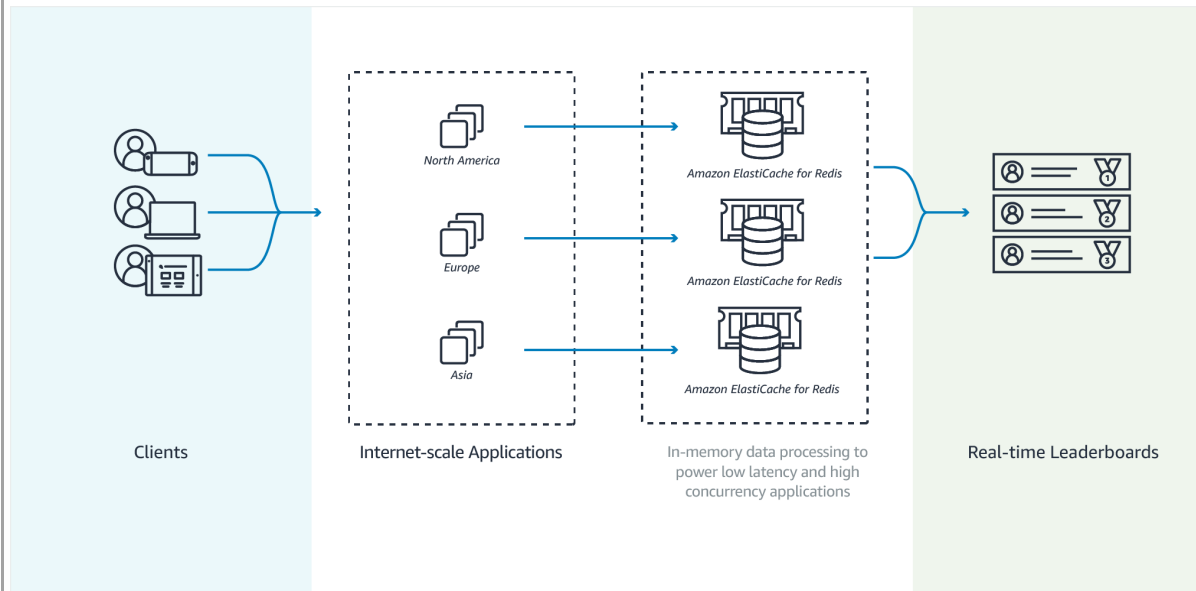
Cancel Save

## FAULT TOLERANT - ELASTICACHE implementation

An ElastiCache Redis cluster provides varying levels of data durability, performance, and cost for implementing disaster recovery or fault tolerance of your cached data. You can choose the following options to improve the data durability of your ElastiCache cluster:

- Daily automatic backups

- Manual backups using Redis append-only file (AOF)
- Setting up a Multi-AZ with Automatic Failover



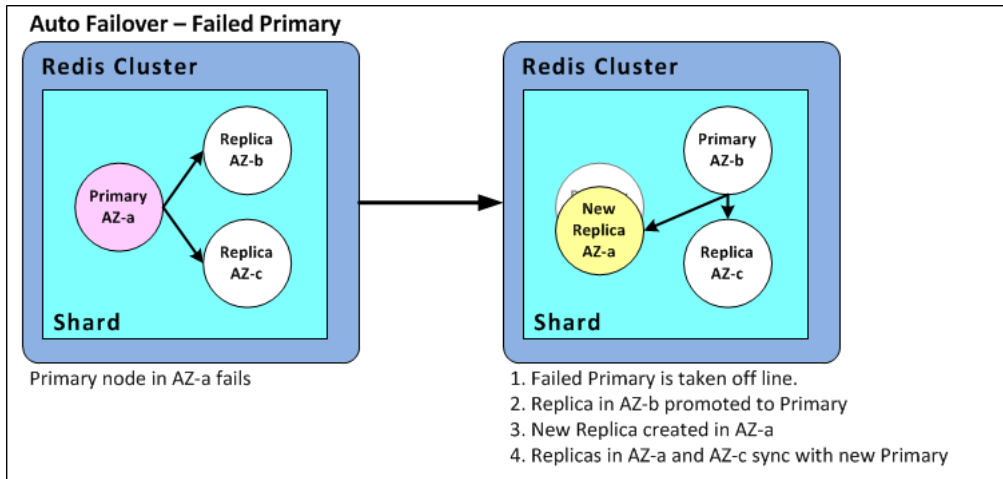
### Redis append-only file (AOF)

- By default, the data in a Redis node on ElastiCache resides only in memory and is not persistent. If a node is rebooted, or if the underlying physical server experiences a hardware failure, the data in the cache is lost.
- If you require data durability, you can enable the Redis append-only file feature (AOF). When this feature is enabled, the node writes all of the commands that change cache data to an append-only file. When a node is rebooted and the cache engine starts, the AOF is "replayed"; the result is a warm Redis cache with all of the data intact.
- AOF is disabled by default. To enable AOF for a cluster running Redis, you must create a parameter group with the **appendonly** parameter set to yes, and then assign that parameter group to your cluster.
- You can also modify the **appendfsync** parameter to control how often Redis writes to the AOF file.

### Multi-AZ with Automatic Failover

- **Multi-AZ with Automatic Failover** provides fault tolerance if your cluster's read/write primary cluster node becomes unreachable or fails. Use this option when data retention, minimal downtime, and application performance are a priority.
- - Its data loss potential is low. Multi-AZ provides fault tolerance for every scenario, including hardware-related issues.

- - Its performance impact is low. Of the available options, Multi-AZ provides the fastest time to recovery, because there is no manual procedure to follow after the process is implemented. Automatic failover buys valuable time that is easily lost when responding to a failure by manually implementing a restore process.
- - Its cost ranges from Low to high. Multi-AZ is the lowest-cost option. Use Multi-AZ when you can't risk losing data as a result of hardware failure or you can't afford the downtime required by other options in your response to an outage.



#### References:

<https://aws.amazon.com/premiumsupport/knowledge-center/fault-tolerance-elasticache/>

<https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/AutoFailover.html>

<https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/RedisAOF.html>

#### Check out this Amazon Elasticache Cheat Sheet:

<https://tutorialsdojo.com/aws-cheat-sheet-amazon-elasticache/>

#### Redis Append-Only Files vs Redis Replication:

<https://tutorialsdojo.com/aws-cheat-sheet-redis-append-only-files-vs-redis-replication/>

#### Comparison of AWS Services Cheat Sheets:

<https://tutorialsdojo.com/comparison-of-aws-services-for-udemy-students/>

#### References:

<https://aws.amazon.com/elasticache/>

<https://docs.aws.amazon.com/AmazonElastiCache/latest/red-ug/elasticache-use-cases.html>

Check out this Amazon Elasticache Cheat Sheet:

<https://tutorialsdojo.com/aws-cheat-sheet-amazon-elasticache/>

## Services

- **CloudSearch**
  - Create on demand Elastic search cluster from offline data
  - Perform search operations
- **Data Pipeline**
- **AWS Glue**
  - ETL tool
- **QuickSight**
  - Business Analytics
- **Athena**
  - Write queries S3 buckets
  - Support of SQL formatted queries on S3 buckets

- AWS analytics engines include CloudSearch, AWS Glue, QuickSight, and Athena
- Analytics engines provide services ranging from search to detailed machine learning
- CloudSearch is a powerful indexing engine within AWS

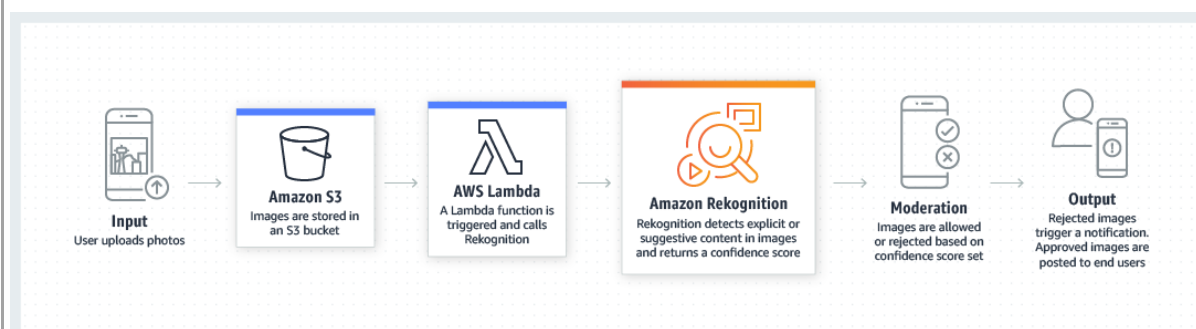
## AWS - Cloud Search

**Amazon CloudSearch** is a managed service in the AWS Cloud that makes it simple and cost-effective to set up, manage, and scale a search solution for your website or application.

With Amazon CloudSearch, you can quickly add rich search capabilities to your website or application. You don't need to become a search expert or worry about hardware provisioning, setup, and maintenance. With a few clicks in the AWS Management Console, you can create a search domain and upload the data that you want to make searchable, and Amazon CloudSearch will automatically provision the required resources and deploy a highly tuned search index.

You can easily change your search parameters, fine-tune search relevance, and apply new settings at any time. As your volume of data and traffic fluctuates, Amazon CloudSearch seamlessly scales to meet your needs.

Amazon Rekognition makes it easy to add image and video analysis to your applications. You just provide an image or video to the Rekognition API, and the service can identify the objects, people, text, scenes, and activities, as well as detect any inappropriate content. Amazon Rekognition also provides highly accurate facial analysis and facial recognition on images and video that you provide. You can detect, analyze, and compare faces for a wide variety of user verification, people counting, and public safety use cases.



**References:**

<https://aws.amazon.com/cloudsearch/>

<https://aws.amazon.com/rekognition/>

**Check out these Amazon CloudSearch and Amazon Rekognition Cheat Sheets:**

<https://tutorialsdojo.com/aws-cheat-sheet-amazon-cloudsearch/>

<https://tutorialsdojo.com/aws-cheat-sheet-amazon-rekognition/>

## AWS - Rekognition

**Amazon Rekognition** can store information about detected faces in server-side containers known as collections. You can use the facial information that's stored in a collection to search for known faces in images, stored videos, and streaming videos. Amazon Rekognition supports the [IndexFaces](#) operation. You can use this operation to detect faces in an image and persist information about facial features that are detected into a collection. This is an example of a *storage-based* API operation because the service persists information on the server.

To store facial information, you must first create ([CreateCollection](#)) a face collection in one of the AWS Regions in your account. You specify this face collection when you call the **IndexFaces** operation. After you create a face collection and store facial feature information for all faces, you can search the collection for face matches. To search for faces in an image, call [SearchFacesByImage](#). To search for faces in a stored video, call [StartFaceSearch](#). To search for faces in a streaming video, call [CreateStreamProcessor](#).

The screenshot shows the Amazon Rekognition 'Facial analysis' demo page. On the left is a navigation menu with options like Metrics, Demos, Object and scene detection, Image moderation, Facial analysis (selected), Celebrity recognition, Face comparison, Text in image, Video Demos, Video analysis, Additional Resources, Getting started guide, Download SDKs, Developer resources, Pricing, FAQ, and Forum. The main content area is titled 'Facial analysis' with the subtitle 'Get a complete analysis of facial attributes, including confidence scores.' It features a large image of a family with bounding boxes around their faces. Below this image are two options: 'Choose a sample image' with two small image thumbnails, and 'Use your own image' with an 'Upload' button, a 'drag and drop' area, and a 'Use image URL' input field with a 'Go' button. On the right, a 'Results' section shows a list of attributes and their confidence scores: 'looks like a face' (99.9%), 'appears to be male' (99.8%), 'age range' (60 - 90 years old), 'smiling' (98.7%), 'appears to be happy' (99.8%), and 'not wearing glasses' (99.8%). There is a 'Show more' link and a 'Request' button. The 'Response' button is highlighted with a red rectangle.

Attribute	Confidence Score
looks like a face	99.9 %
appears to be male	99.8 %
age range	60 - 90 years old
smiling	98.7 %
appears to be happy	99.8 %
not wearing glasses	99.8 %

AWS Storage Gateway offers file-based, volume-based, and tape-based storage solutions. With a tape gateway, you can cost-effectively and durably archive backup data in GLACIER or DEEP\_ARCHIVE. A tape gateway provides a virtual tape infrastructure that scales



seamlessly with your business needs and eliminates the operational burden of provisioning, scaling, and maintaining a physical tape infrastructure.

You can run AWS Storage Gateway either on-premises as a VM appliance, as a hardware appliance, or in AWS as an Amazon Elastic Compute Cloud (Amazon EC2) instance. You deploy your gateway on an EC2 instance to provision iSCSI storage volumes in AWS. You can use gateways hosted on EC2 instances for disaster recovery, data mirroring, and providing storage for applications hosted on Amazon EC2.

**References:**

<https://docs.aws.amazon.com/rekognition/latest/dg/collections.html>

<https://aws.amazon.com/storagegateway/file/>

**Check out this Amazon Rekognition Cheat Sheet:**

<https://tutorialsdojo.com/aws-cheat-sheet-amazon-rekognition/>

**Tutorials Dojo's AWS Certified Solutions Architect Professional Exam Study Guide:**

<https://tutorialsdojo.com/aws-cheat-sheet-aws-certified-solutions-architect-professional/>

**IN-CORRECT :** although this is using the right combination of AWS Storage Gateway and Amazon Rekognition, take note that you can't directly fetch the media files from your tape gateway in real-time since this is backed up using Glacier. Although the on-premises data center is using a tape gateway, you can still set up a solution to use a file gateway in order to properly process the videos using Amazon Rekognition. Keep in mind that the tape gateway in AWS Storage Gateway service is primarily used as an archive solution.