

Strengths and weaknesses of databases cont'd**3. Key-value store databases**

A **key-value store database** is a type of nonrelational **database** that uses a simple **key-value** method to **store** data. In key-value, each value is associated with a specific **key**. A **key-value database stores** data as a collection of **key-value** pairs where the **key** serves as a unique identifier. Both **keys** and **values** can be anything, ranging from simple objects to complex compound objects. It just has a one-way mapping from the key to the value to store data.

A key-value store uses an array of keys where each key is associated with only one value in a collection. It is quite similar to a dictionary or a map data structure. It's also known as an associative array.

- “key” is a unique identifier associated only with the value. Keys can be anything allowed by the DBMS.
- “Values” are stored as blobs and don't need predefined schema. They can take nearly any form: numbers, strings, counters, JSON, XML, HTML, PHP, binaries, images, short videos, lists, and even another key-value pair encapsulated in an object.

Strengths

- Are very very flexible and able to handle a very wide array of data types easily.
- Have high performance as Keys are used to go straight to the value with no index searching or joins.
- High performance because the integrated caching feature allows users to store and retrieve data in the shortest time possible.
- Can be moved from one system to another without rewriting code. i.e. easily Portable

- easy to scale without disrupting operations. Users can add and remove servers depending on their needs without causing undesirable disruptions.
- users can simply add new features when the need arises

Weaknesses

- No query language to retrieve data. Only have some simple operations such as get, put and delete.
- Almost impossible to query values, because they're stored as a blob and can only be returned as such.
- difficult to do reporting or edit parts of values.
- Data querying (retrieving) usually handled manually at the application level.

Final Verdict

Based on the various parameters, strengths and weaknesses, a Key-value Store database may be used for

- Session management at high scale
- User preference, profiles and settings
- Unstructured data such as product reviews or blog comments
- Product recommendations; latest items viewed on a retailer website drive future customer product recommendations
- Ad servicing; customer shopping habits result in customized ads, coupons, etc. for each customer in real-time
- Can effectively work as a cache for heavily accessed but rarely updated data i.e. Data that will be accessed frequently but not often updated

Some of the examples of Key-Value Store databases are Amazon DynamoDB, MemcacheDB, InfinityDB, Redis, Aerospike, Oracle Berkeley DB, Riak KV, Voldemort etc.

Assignment

1. What are the benefits of Key-Value Store? Where these databases can be used?
2. In what situations, key-value store databases are not suggested?