

RedisBloom

SILICON VALLEY ENGINEER

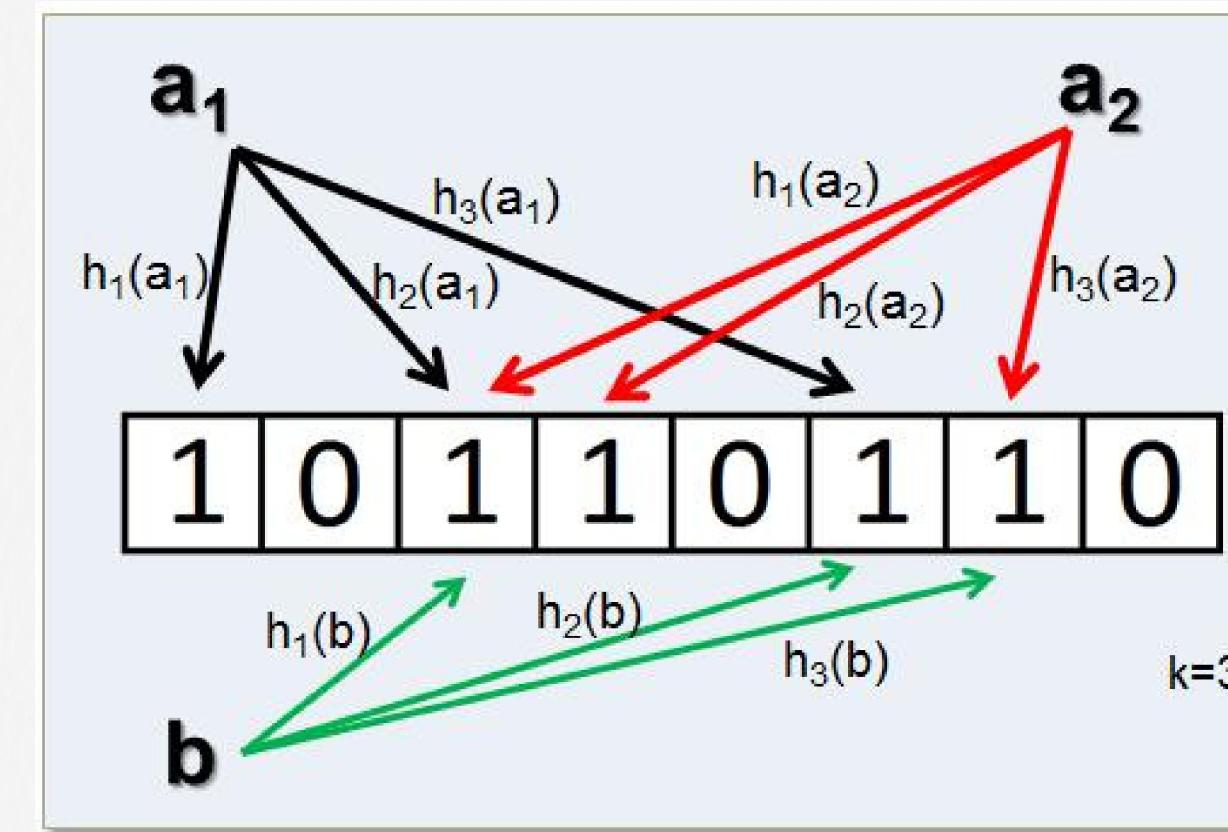
- RedisBloom introduces **probabilistic data structures to Redis**, including Bloom filters, Cuckoo filters, Count-Min Sketch, and others.
- These structures allow for **highly efficient testing of membership and frequency estimation with a trade-off in accuracy defined by a small probability of false positives**.
- RedisBloom is useful for applications where **space efficiency** and **query speed are critical**, such as caching, deduplication, and real-time analysis tasks.



Bloom Filter

SILICON VALLEY ENGINEER

- **Structure and Operation:** A Bloom filter uses multiple hash functions to map elements to several positions in a bit array. When adding an element, it sets the bits at these positions to 1. To check membership, it checks the bits at the positions determined by the hash functions; if all are 1, the element is possibly in the set; if any bit is 0, the element is definitely not in the set.



Bloom Filter

SILICON VALLEY ENGINEER

- **Advantages**

- Space-efficient for representing a set.
- Adding elements is very fast.

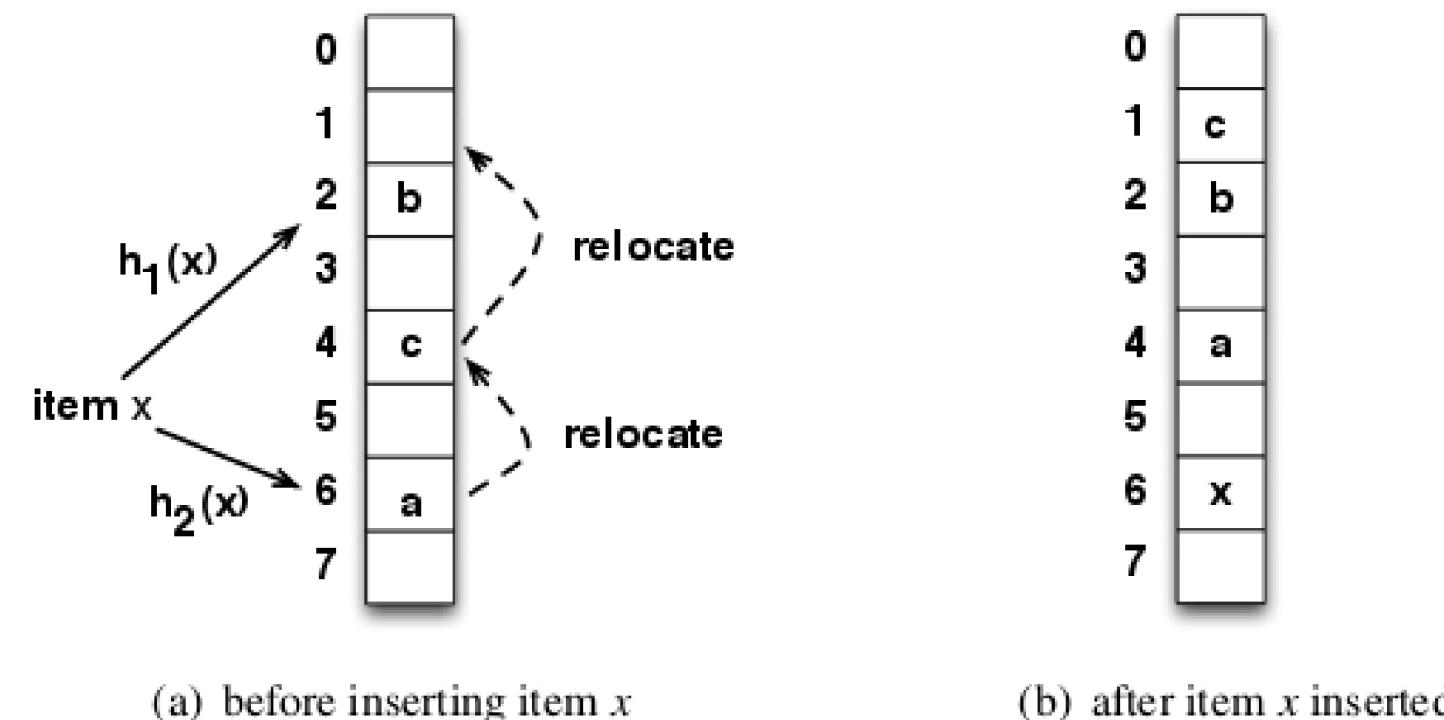
- **Disadvantages**

- False positives are possible (it might say an element is in the set when it's not), but false negatives are not (if it says an element is not in the set, it's definitely not).
- The false positive rate increases as more elements are added.
- It does not support deletion of elements (although variants like Counting Bloom filters do).
-

Cuckoo Filter

SILICON VALLEY ENGINEER

- **Structure and Operation:** A Cuckoo filter is based on the cuckoo hashing technique. It stores each element in a bucket with a limited capacity, and uses a pair of hash functions to determine the bucket locations. If a new element needs to be added to a full bucket, it can displace an existing element to its alternate location, potentially causing a chain of displacements.



Bloom Filter

SILICON VALLEY ENGINEER

- **Advantages**

- Like Bloom filters, Cuckoo filters are space-efficient and support fast membership queries.
- They support deletion of elements, unlike basic Bloom filters.
- They generally provide a lower false positive rate than Bloom filters for a given space efficiency.
- They can be more efficient in dynamic scenarios where elements are added and removed over time.

- **Disadvantages**

- The implementation of Cuckoo filters is more complex due to the handling of displacements.
- They may not be as efficient as Bloom filters in static scenarios where the set of elements does not change, due to the overhead associated with displacements.

Bloom Filter vs. Cuckoo Filter

SILICON VALLEY ENGINEER

- **Use a Bloom filter when you have a static set of elements**, or if space efficiency and insertion speed are your primary concerns, and you can tolerate a certain rate of false positives without the need for element deletion.
- **Use a Cuckoo filter if you need a dynamic set** where elements will be added and removed over time, require lower false positive rates, or need the ability to delete elements.

◦