

# Simplified data partitioning in a consistent hashing based sharding implementation

---

By Pulak Sahoo

M.Tech(1st year)- 182IT016

Existing Solution

v/s

Present Solution

# Existing Solution

—

# Operations

Found in the algorithm.

- Creation of virtual shards
  - Distributing keys to the virtual shards
  - Hash ring implementation
  - Allocating the virtual shards to the servers
-

# Procedure

```
function CreateShard (S, V)
    virtual partition size  $\leftarrow$  key space/V
    cnt  $\leftarrow$  0
    for i = 1 to n do
        for j = 1 to v do
            LB[i][j]  $\leftarrow$  cnt * virtual
            partition size
            cnt  $\leftarrow$  cnt + 1
        end for
    end for
    return LB
end function
```

▷ S[1..n] - Servers in the partitioning topology, V - Number of Virtual Shards

▷ Calculate size of each virtual partition

▷ LB[1..n][1..v] - Lower Bound for each server

# Present Solution

---

# Operations

Found in the algorithm

- Threshold ratio Calculation for every server
  - Keyspace division for every server
  - Allocation of keyspace to each server
-

# Procedure

function CreateShard(S, T, KS)

$T_{sum} \leftarrow \sum_{i=1}^n T_i$

  for  $i = 1$  to  $n$  do

    key space ratio  $\leftarrow T_i / T_{sum} \times 100$

    shard size  $\leftarrow$  key space ratio / 100  $\times$  KS

    if  $i = 1$  then

      LB[i]  $\leftarrow$  shard size

    Else

      LB[i] = LB[i - 1] + shard size

    end if

  end for

  return LB

end function

▷ S-Servers, T-Thresholds, KS-key space size

▷ Sum of Thresholds

▷ Percentage of key space allocated

▷ key space size allocated

▷ First shard



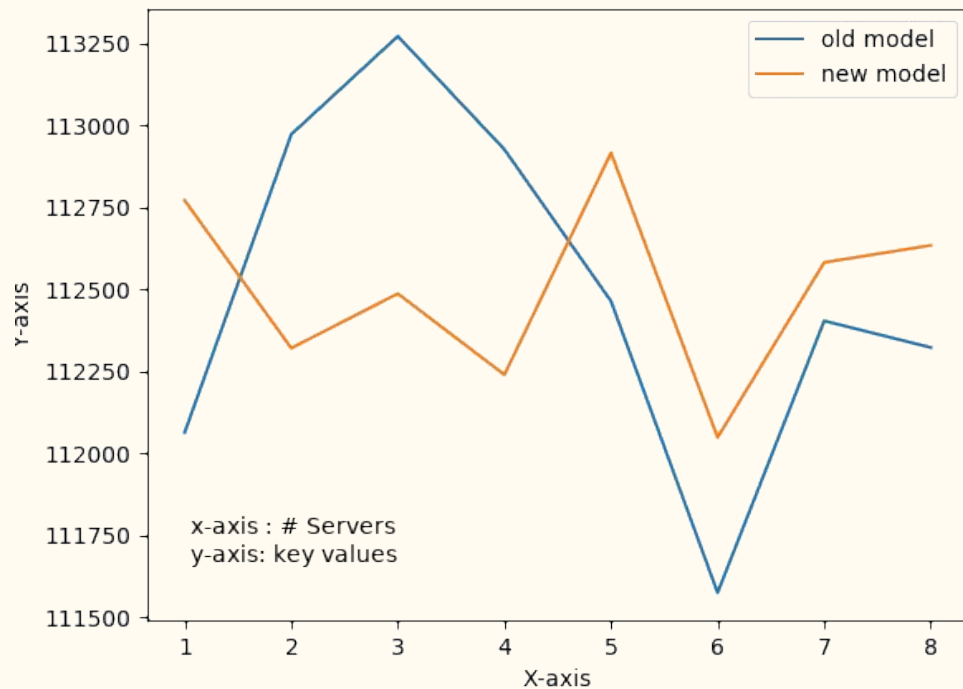
# Comparison

—

# Data distributed among 8 servers

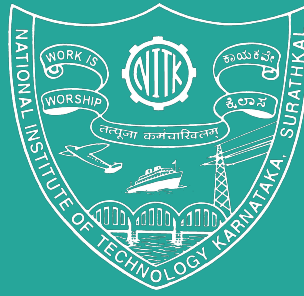
Servers	Existing Algorithm	New Algorithm
S1	112064	112771
S2	112972	112321
S3	113271	112487
S4	112926	112240
S5	112464	112916
S6	111576	112049
S7	112404	112582
S8	112323	112634

# Comparison between the two Algorithms



# Conclusion

The paper proposes an alternative to creating and maintaining shards, without the use of virtual shards. The paper also verifies the efficiency of the proposed method, for both creating and maintaining shards, and finds that the proposed method performs satisfactorily and in some cases better than the existing methods that use virtual shards.



# Thank You!

—