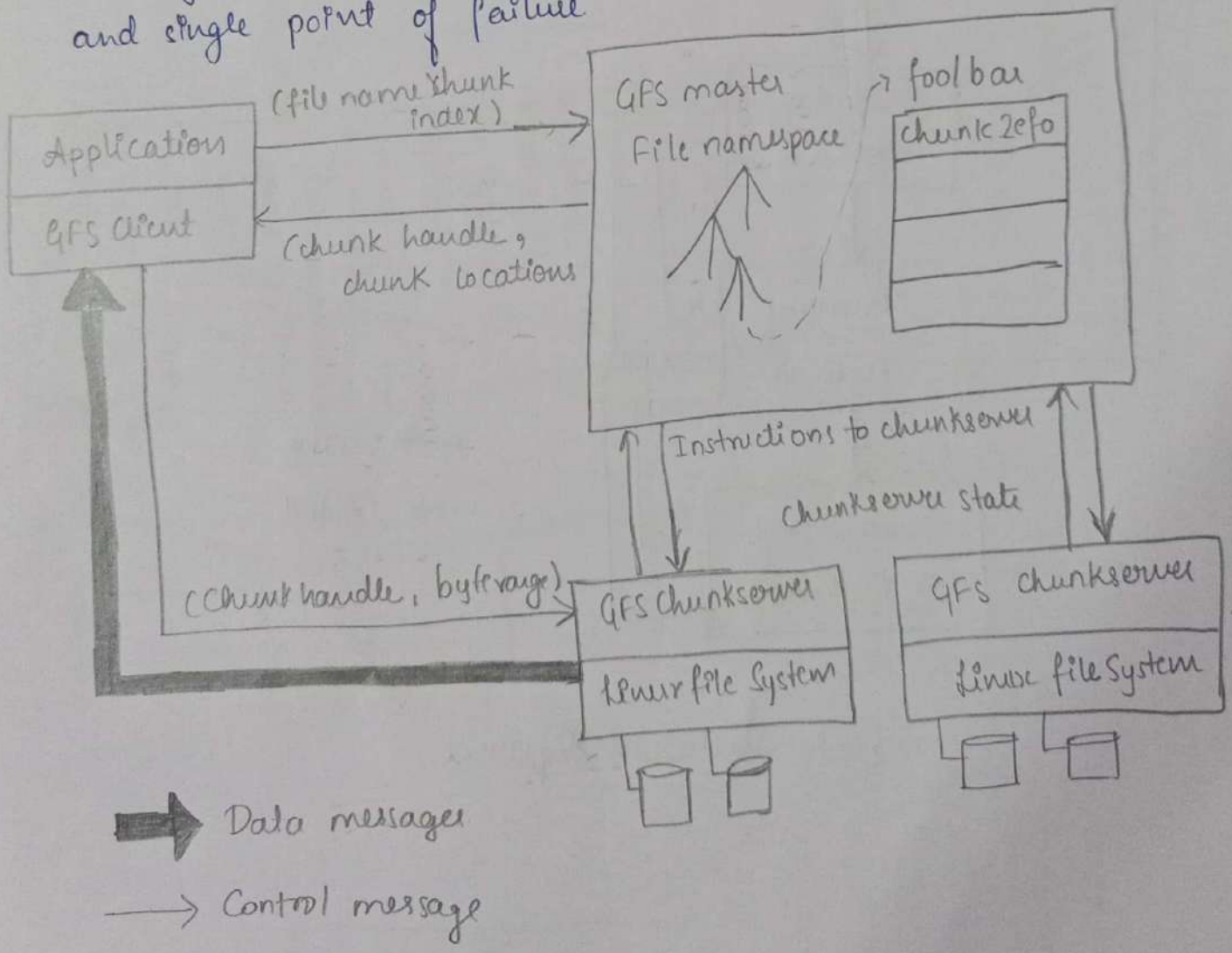
Google file System

→ In GFS, there is a single master in the whole cluster which stores metadata of the nodes act as the chunk servers for storing data

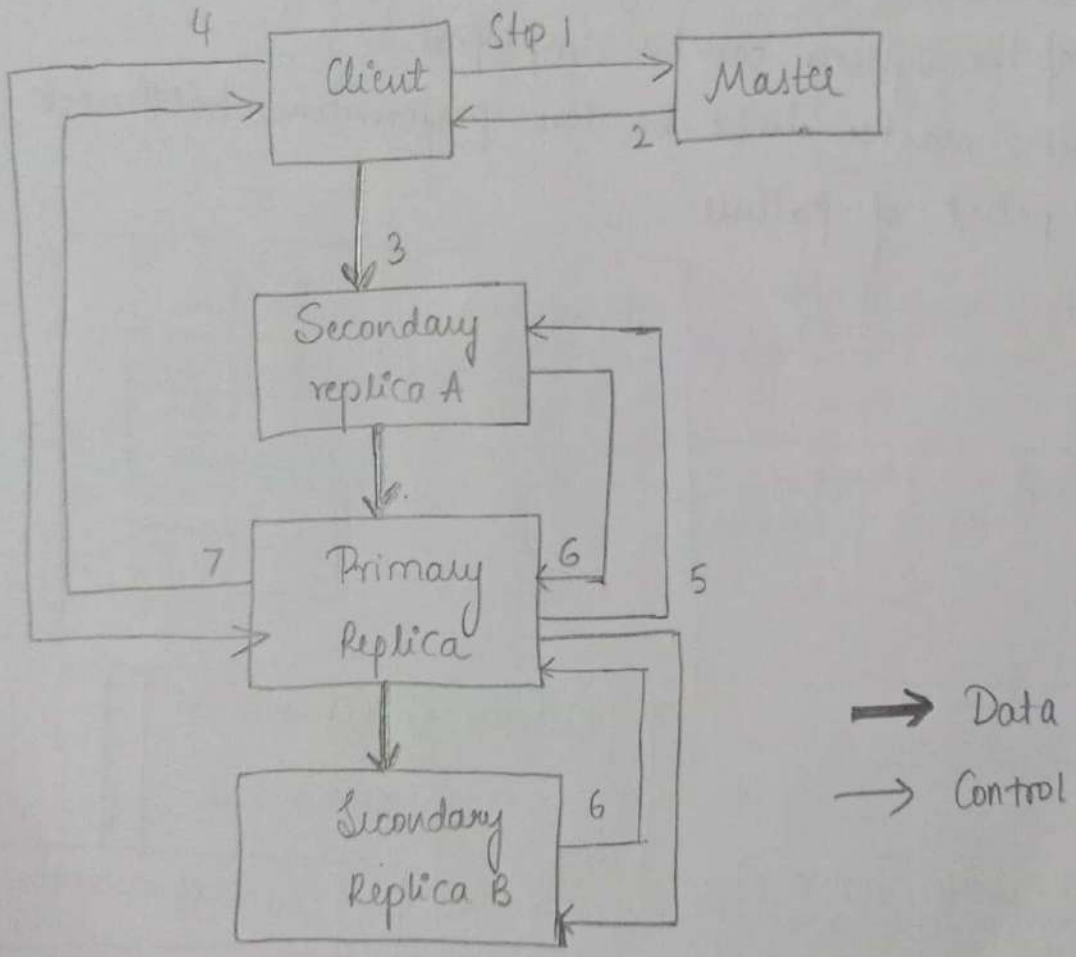
→ The file system namespace and locking facilities are managed by master. The master periodically communicates with chunk servers to collect information and give instruction to chunk server to do fail recovery or load balancing. A single master with many complicated distributed algorithms can be avoided and the design of the system can be simplified.

→ The single GFS master could be the performance bottleneck and single point of failure

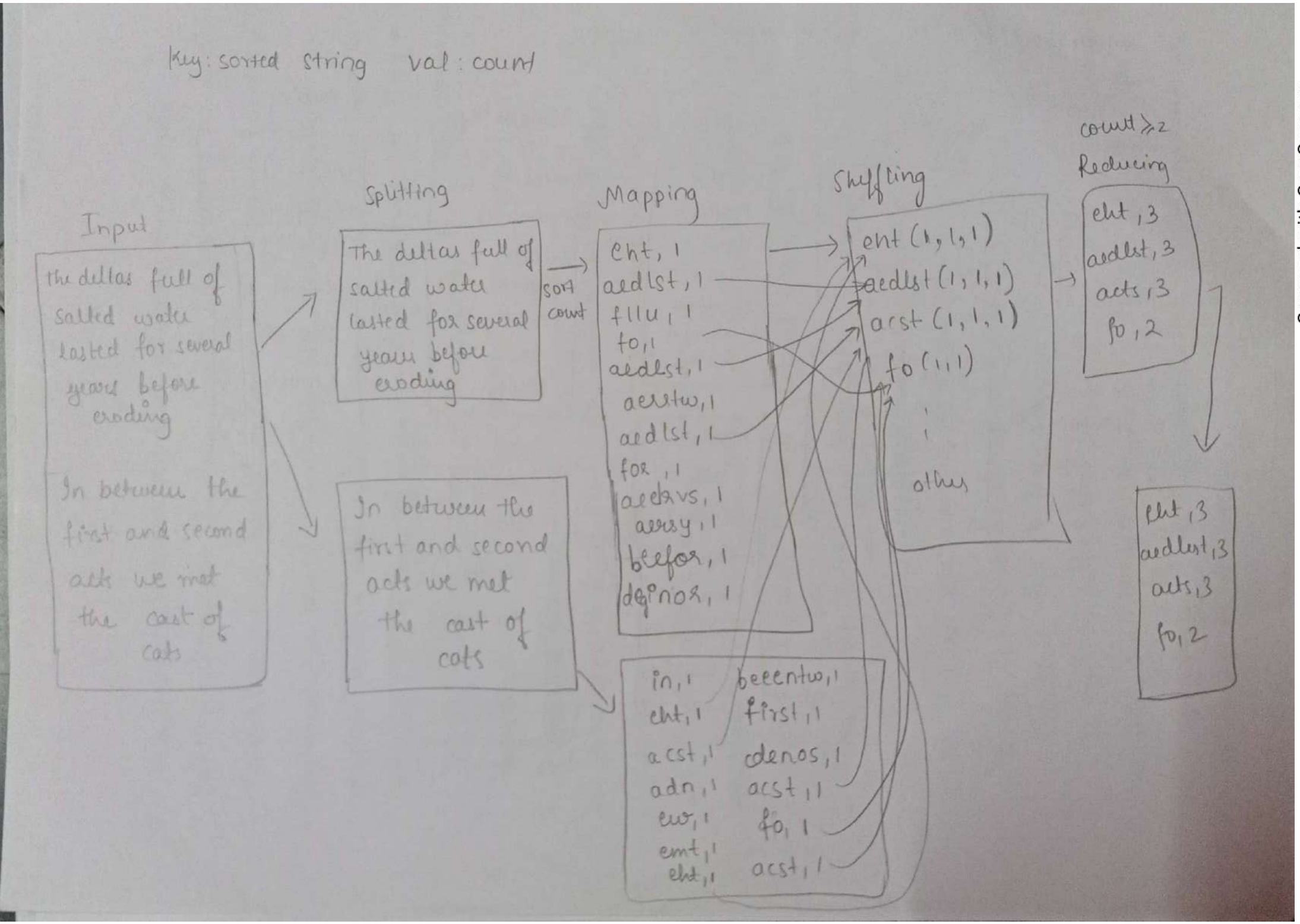


→ To mitigate this, Google uses a shadow master to replicate all the data on the master and the design guarantees that all data operations are transferred between the master and clients and they can be cached for future use.

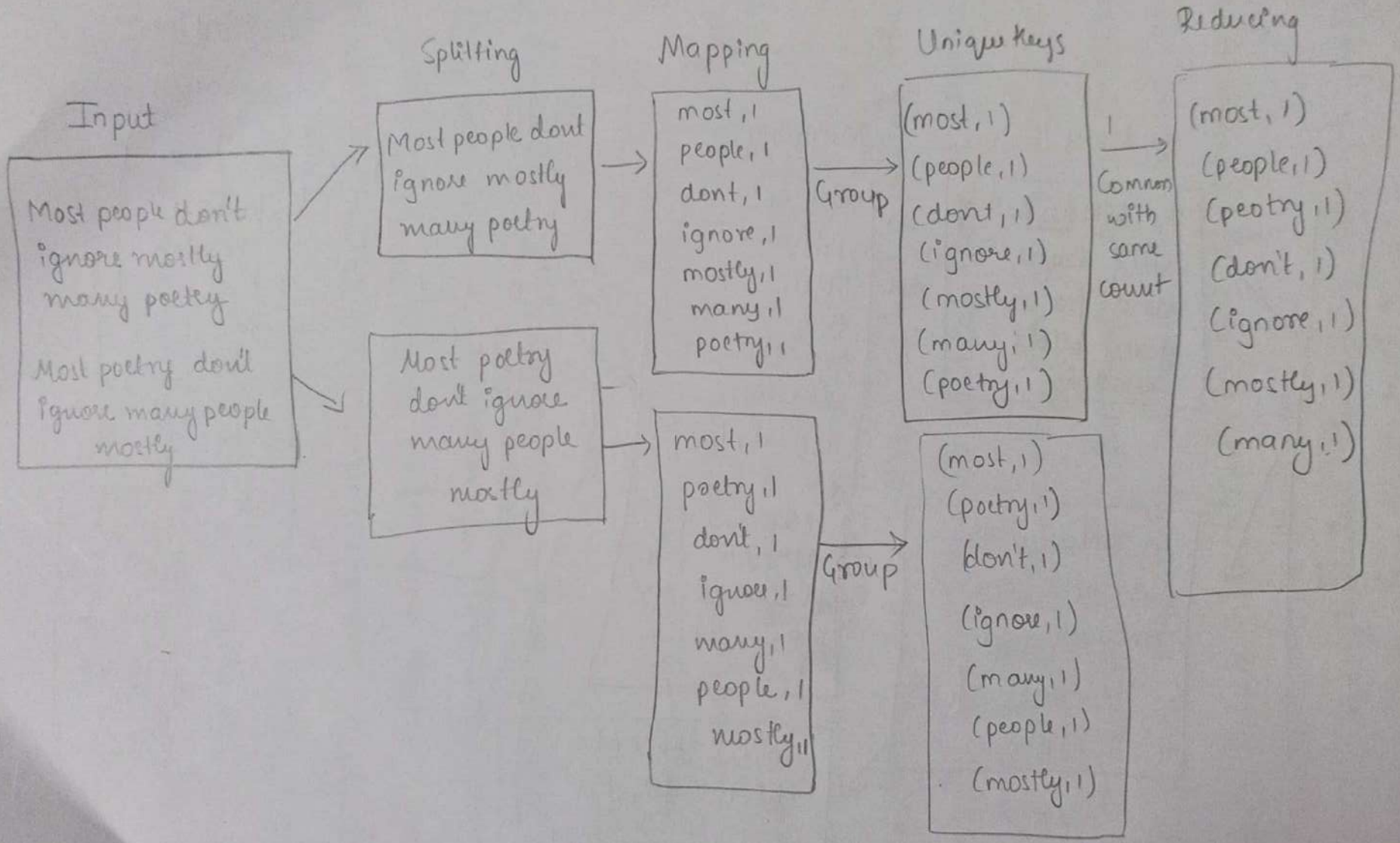
→ The current quality of commodity servers, the single master can handle a cluster more than 1000 nodes.



Data Mutation Sequence.



Key: word in a file value: count of occurence



Characteristics And Advantages of Borrowed Virtual Time?

→ Conventional schedulers are bad for real time tasks.

→ Priority schedulers- If there was a high priority or greedy thread / source it would overconsume CPU and bandwidth.

→ In deadline based schedulers requires applications to predict whether and when they will need CPU time in advance. So in case of overestimation of CPU needs they may get refused or if they underestimate they may miss a deadline.

Borrowed Virtual Time

→ It supports both real time and best-effort type of tasks. It is simple for programmers to use. It is simple and cheap to implement

→ Monitoring virtual time a_i and schedule the process w.r.t. lowest effective virtual time. It helps in scheduling a mix of applications

→ Time warp speeds up execution of processes and threads. It is an allowing factor which allows to borrow against its future CPU usage or bandwidth. So we can borrow from future allocation in case of emergency in advance. Consider w_i is timewarp

It gives schedulers / thread a dispatch preference whenever it gets scheduled. It does not affect long term share of its CPU weight and usage.

→ It affects its priority on wakeup.

→ It supports low latency - dispatch, real time, weighted-sharing of CPU resources and obeys soft and hard real time constraints

→ Consider thread i with effective virtual time Ei and Ai - actual virtual time. wi - time warp

→ Each thread has its own SVT (Scheduler virtual time) i.e minimum amount of virtual time

→ threads are dispatched according Ei This policy is called as "Earliest Virtual Time".

→ There will be context switching b/w threads and is triggered by:

(i) Blocking of running thread

(ii) Interrupt

(iii) time quantum expires

(iv) If thread becomes runnable after its sleeping period.

→ Context switch occurs from i to j if $Aj \leq Ai - \frac{C}{wi}$

→ wi allows to borrow ei from its future allocation

* $Ei \leftarrow \begin{cases} Ai & WB = OFF \\ Ai - wi & WB = ON \end{cases}$

* If thread Ji becomes runnable after sleep its $Ai \leftarrow \max[Ai, SVT]$ doesn't allow thread to sleep for longer time. Each thread has constant Ki.

* ↑Wa more consumption of ei and vice-versa

* $Ai \leftarrow Ai + \Delta$, Δ = time advance, $\Delta = \frac{Ka}{wa} = \frac{Kb}{wb}$

Advantages Of Cloud

1) On-demand :- Resources are utilized only on demand for that period of time and pay for what we use.

2) Availability of cloud services 24x7 and can be accessed across the globe

3) Easy maintainenace : Easy to maintain, need not worry about backup, safety is also provided for the data and easy to use

4) Scalability : It is scalable to meet unpredicted needs.

5) Elasticity : Rapid elasticity is provided by cloud ; sudd-enly requirement changes then CSProvides facility of provisioning resources when and where required properly

6) Economical : Cheap and pay-as-yo-go model

7) No hardware required : Since everything is hosted on cloud, a physical storage center is no longer needed

8) Efficient : Cloud provides

Describe types of Virtualization

Types of virtualization

i> Network : The available resources / bandwidth is split in such a way that it is given to different channels which appears to be separate and distinguished. Multiple networks on same LAN. Each user gets different network. Virtual IP's is given to each user.

ii> Storage : Grouping multiple data storages to appear as one unit. Pooling all storage space from several interconnected device and simulated as single storage device. This is now managed with single console which gives single view of different storage space as large single storage view. Used in storage area network . Eg: For backup, archiving of data and data recovery. From one console all diff storage spaces is handled

iii> Server: It masks server - resource. It simulates a physical server and gives to user. Each user feels he has own dedicated user. User need to managing of complex resources. It increase resource utilization.

iv) Data : Data management (it's location, metadata, format) is abstracted and complexity is hidden and made available to user. User is freed from data management. Used for business purpose.

v) Destop : Use desktop from anywhere. Workstation is running. Provides portability. It can access from anywhere and also modify i.e create, delete, save folders. Backend is same.

vi) Application virtualization : Application is encapsulated and is relieved from dependency of underlying hardware. "Write Once Run Anywhere" Codes.

Q) Security risks posed with shared images?

Even, when we assume that CSP is trustworthy, many users either ignore or underestimate danger possessed by other source of concern. one of them critical to IaaS cloud delivery model, i image sharing

for eg: An AWS user has option to choose b/w AMI's accessible through quick start or community AMI ... of EC2 service.

→ To create AMI, we can start from running system, from another AMI or from Image of VM, copy contents to S3 buckets. Step1 is to create an Image. Step 2: Compress and encrypt the image Step 3: Split the image into several segments and upload to S3

→ To use an AMI, user has to specify resources, provide credentials for login, provide a firewall configuration, specify the region.

→ Many of images analyzed by recent report allowed access to undelete files, recover credentials, private keys with standard tools

→ A software vulnerability audit revealed that 98% of windows AMI's and 58% of linux AMI's had critical vulnerabilities

→ Security risks: i) Unsolicitated connections

ii) Malware

iii) backdoors and left over credentials.

Features of Google File System:

1) It was designed for high fault tolerance.
2) Master and chunk servers can be restarted in a few seconds and with such a fast recovery capability, window of time in which is unavailable can be greatly reduced
3) Each chunk is replaced at least 3 places and can tolerate at least two data crashes for a single chunk of data
4) GFS can achieve the goals of high availability, performance and implementation.
5) The shadow master handles the failure of the GFS master
6) It demonstrates how to support large scale processing workloads on commodity hardware designed to tolerate frequent component failure optimized for huge files that are mostly appended and read
7) For data integrity, GFS makes checksums on every 64KB block in each chunk.