Architechture

Worker

Communication Handler

Worker <-> Master

Worker <-> Worker

Memory Handler

Partitioning Unit

Sort Unit

Sort Unit

Merge Unit

Command handler/ Main worker Unit...

Master

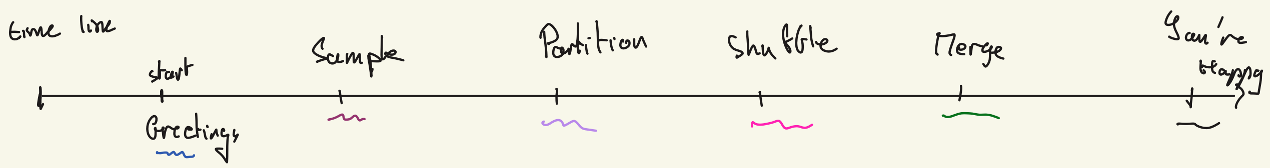
Sampk

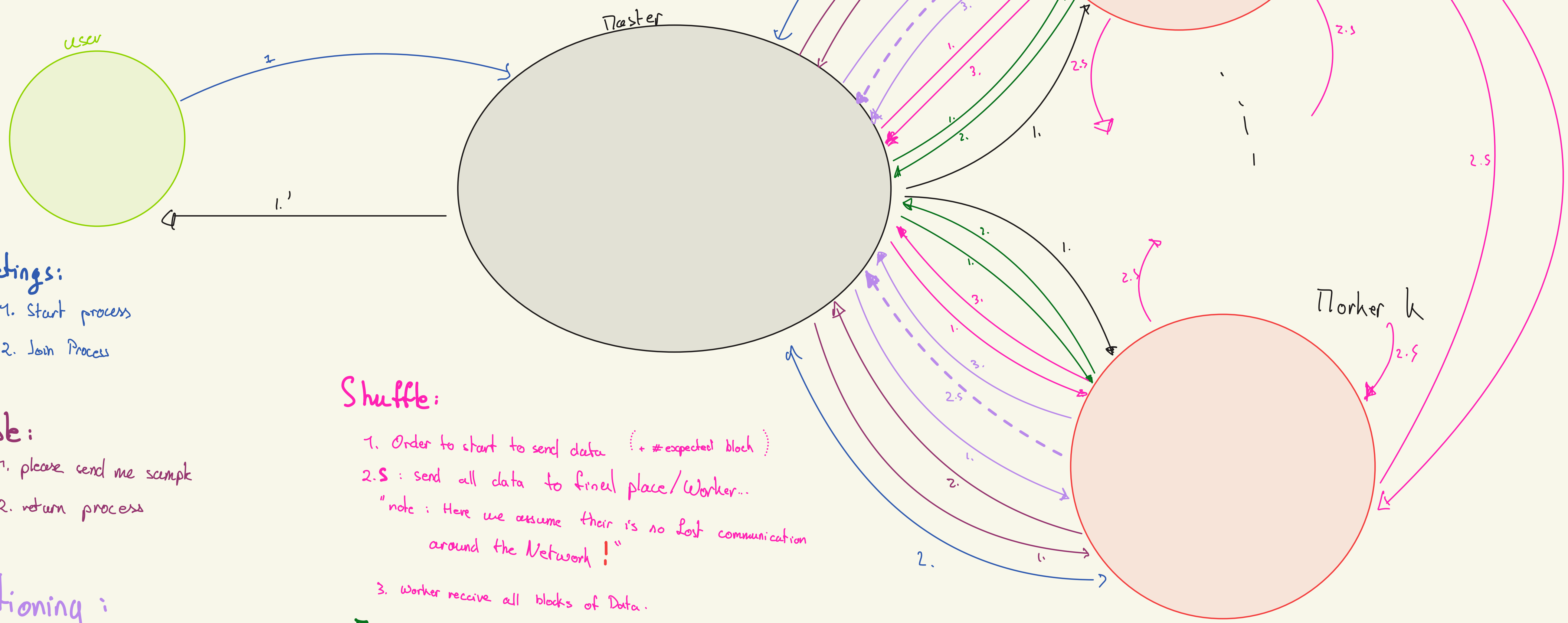
Network / Worker manager

State Manager

User Main Interface

History / Logger / Transaction history





Greetings:

1. Start process
2. Join Process

Sample:

1. please send me sample
2. return process

Partitioning:

1. Send the range of partitioning + please start to partition you're stuff!

2. S*: Send Meta-data each time you're <1. partition; 2. sort; store in temp. file> a new block of data
(*S: many blocks will be created)

3. worker has finish to partion the memory

Shuffle:

1. Order to start to send data (+ #expected block)
2. S: send all data to final place/Worker..
"note: Here we assume their is no lost communication around the Network !"
3. Worker receive all blocks of Data.

Merge:

1. Start Merging
2. Worker finish to merge

End:

1. & 1.' : end of the process.

Worker Requierments

Communication Handler

Worker <-> Master | Worker <-> Worker

Memory Unit

Partitioning Unit

Sort Unit | Sort Unit

Merge Unit

Worker Management Unit

"Data Processing Unit"

- = store data
- = title
- = feature/methode to handle

Communaction Handler :

- own ip

Worker -> Master

- master ip
- receive order
- send meta-data
- send sample
- send state of computer
- send join request

Worker -> Worker

- Could store the slice-ip (ip from other worker & the range corresponding)
- Send Datablocks
- Receive Datablocks

Memory Unit:

- store data & meta-data information
- load full chunchs
- slow read data chunchs
- find file
- create temp file
- write temp file
- write final file

Sort Unit:

- ??
- take one full data-chunks
  1. sort it
  2. store & more in local memory
  3. ??

Merge Unit

- slow read all usefull temp file, then merge them into the final result.

Worker Management Unit

- create process
- answer master request
- (manage all the worker action)
- Handle every request from outside.

Partitioning Unit.

- partition the memory into many unsorted data-array

Master

Sampk

Network / Worker manager

State Manager

User Main Interface

History / Logger / Transaction history

Sampler

- the sample receive
- Sort the sample
- Compute the range

Network Manager

- the ip of all the actor
- send the request
- receive workers informations
- communicate with the other part of the master

State Manager

- current state
- previous state
- worker state ...
- manage the worker depending of the current state
- change of state (update) when the condition are fullfilled!

User Main Interface

- handler of the request of the user,
- Send message to the user
- start / kill the process
  - ↳ interract with process

Logger / History of Translation:

"keep in memory every transact°"

→ to later for fast recovery.