Today

Internet scale Computing
(Lesson 9)

✔ * giant scale services

✓ * Map-reduce programming model
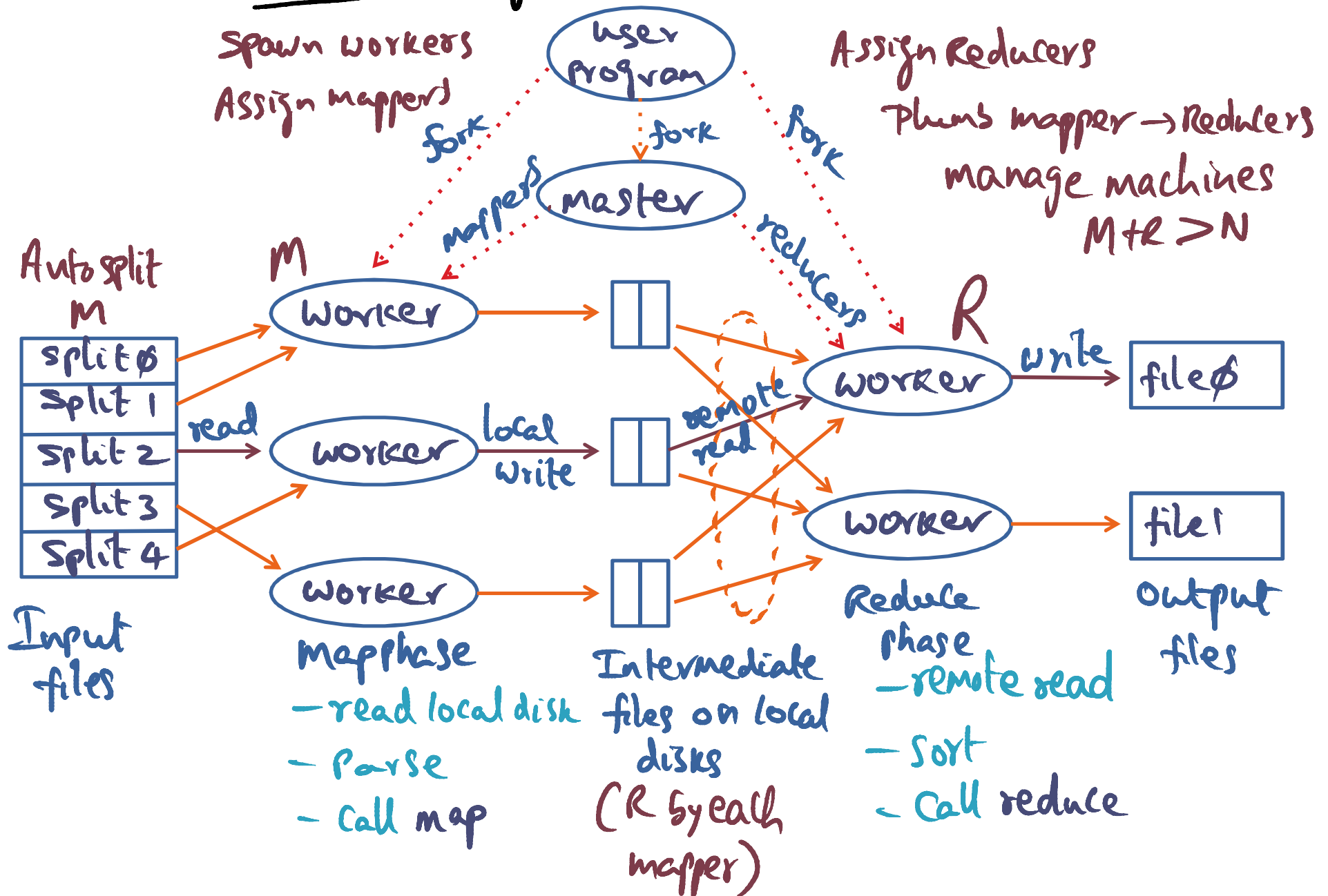
⇒ * CDN (coral)

Next week

Real time (Lesson 10)

* Please watch videos before class *

Heavy Lifting Done by the runtime

Spawn workers
Assign mappers

Assign Reducers
Plumb mapper → Reducers
manage machines
$M + R > N$

user program

fork

master

fork

fork

mappers

reducers

Auto split
M

Input files

M

Split 0
Split 1
Split 2
Split 3
Split 4

read

Worker

Worker

Worker

Mapphase
- read local disk
- Parse
- Call map

local write

Intermediate files on local disks
(R by each mapper)

remote read

R

Worker

Worker

Reduce phase
- remote read
- sort
- Call reduce

write

file 0

file 1

output files

Issues to be handled by the runtime

Master data structures
- location of files created by completed mappers
- scoreboard of mapper/reducer assignment
- Fault tolerance
  - # start new instances if no timely response
  - # completion message from redundant stragglers
- Locality management
- Task granularity
- Backup tasks

Key takeaways

- map-reduce => simplicity
- domain expert's programming burden?
  - write "map" and "reduce" functions
- heavy lifting under the cover by the runtime system