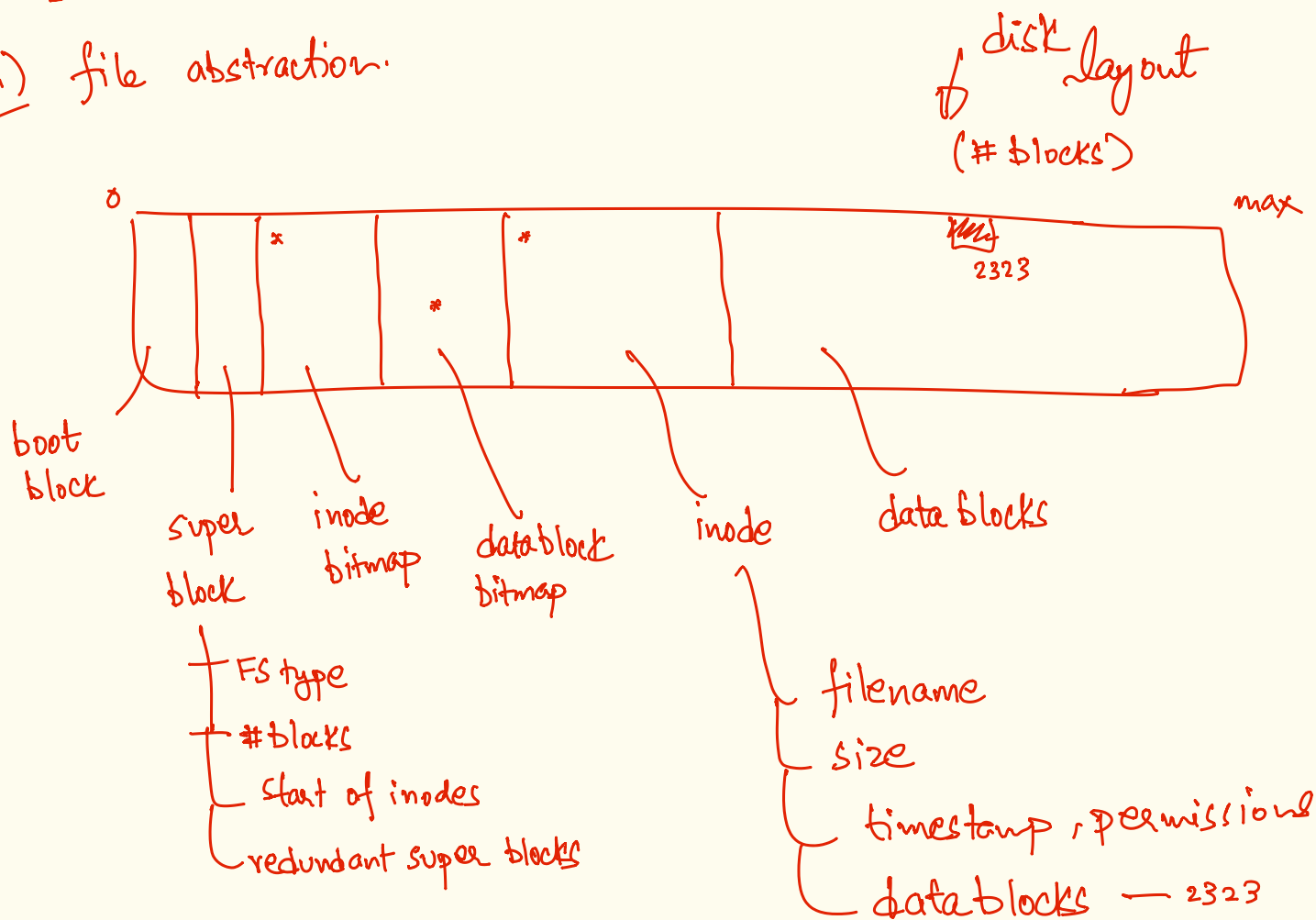


storage:(i) file abstraction.(ii) object store

— serialized objects/variables stored on disk.

```
~ struct student {
    int roll no.;
    char [50] name;
    ...
    }
    filename
```

~ s.serialize (filename);

~ s = deserialize (filename);

(iii) Key-value store

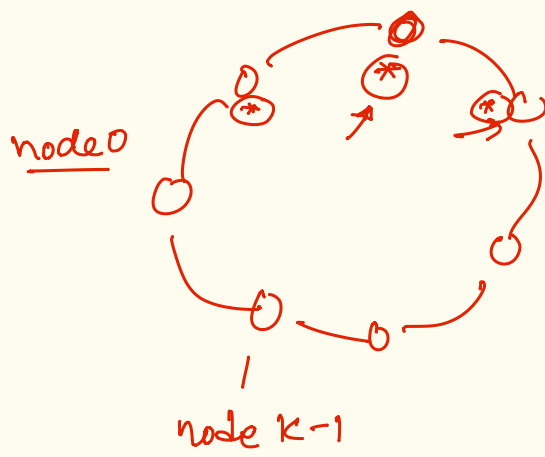
put (Key, value);

v = get (Key);

⊛ ~ distributed KV stores

⊛ how to distribute Keys?

⊛ # consistent hashing.



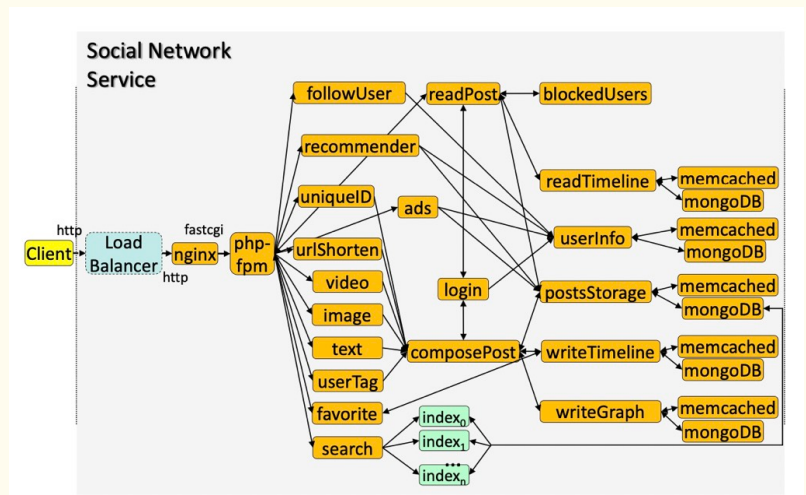
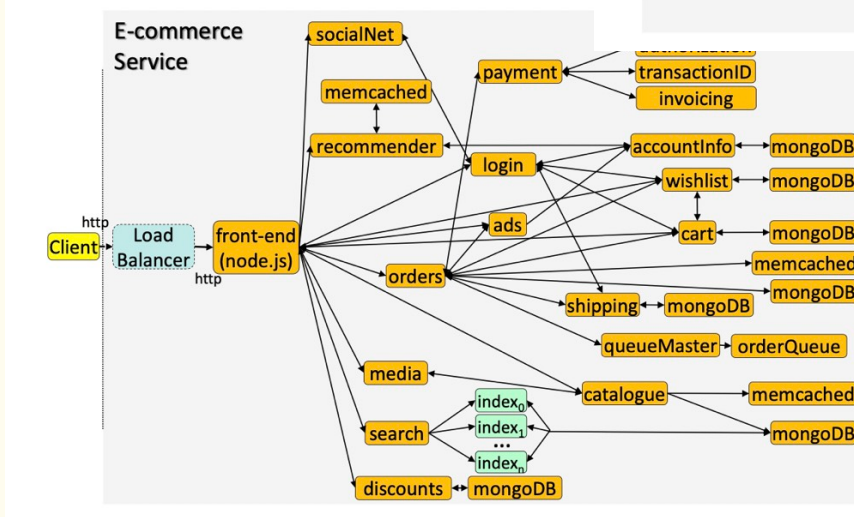
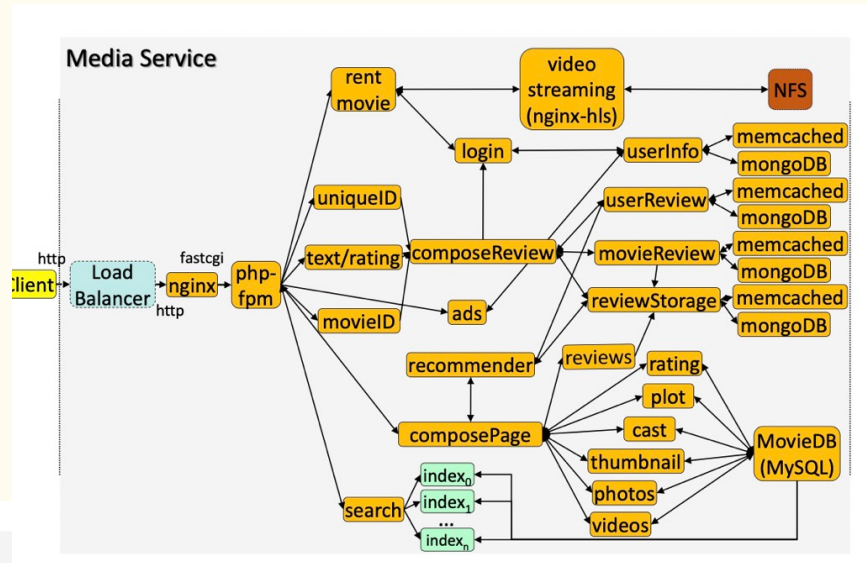
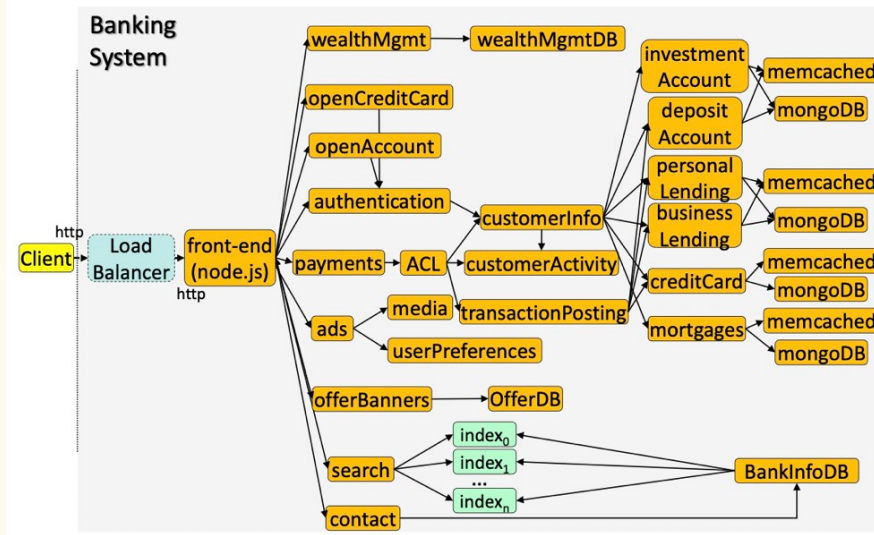
servers - 0 — k-1

(Keys) % k

to identify

server
responsible for key.

DeathStarBenchmark



★ Internet-scale of distributed systems

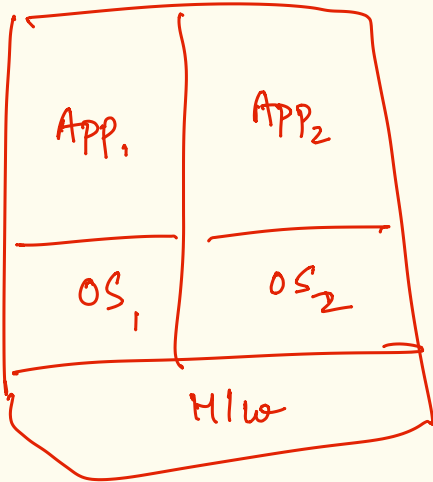
cloud computing

compute-as-a-service

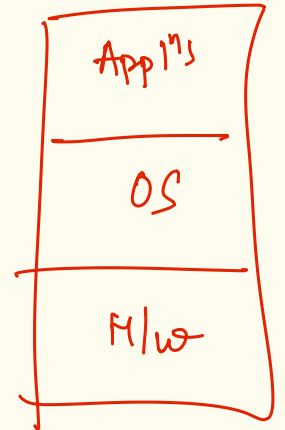
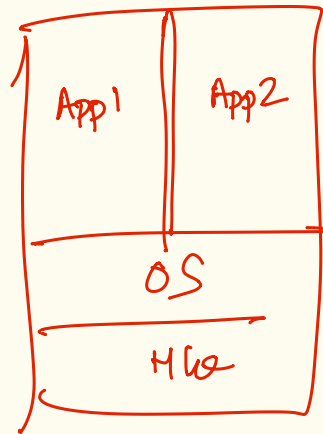
machines, platforms, IDEs, tools, slw, hardware...

cloud computing
systems infrastructure

(i) virtual machines



(ii) containers



process-view

- ~ address space
- files & directories
- ~ Network interface endpoints
↳ eth0

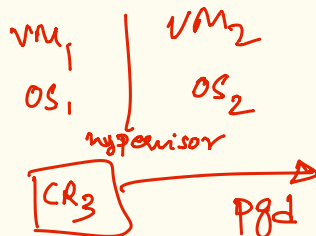
OS system-view

- ~ registers
- ~ h/w ISA/interfaces devices
- ~ phy. address range

VM is an abstraction of the OS-view

VMM
hypervisor

★



chroot ~ change the root directory.

namespaces ~ demarcating FS views.
resource

cgroups

control group of processes
extent of memory scheduling # CPUs
+ ps
+ net
+ pid

~ docker, runc ~ container instantiation tool

~ K8S ~ container orchestration tool.

Summary of CS 744

