

design principles of distributed systems.

ideas
issues

- ① Endsem exam
- 21st Nov. 1:30pm(?)
- ② Project
- 24/25/26/27(?) Nov.
- demo, report, {documentary, storybook, movie...}

Q what is a distributed system?

- collection (of interconnected) computational resources
- common task/activity/requirement

reqs. of any computing system

- correctness
- performance
- failure handling

examples:

- ① web-based services
- infrastructure-as-a-service
(compute infra)
└ VMs
└ storage

- banking
- multimedia
- ecomm
- social n/w
- asc

② OS-level type abstractions

- ~ DFS
- ~ Distributed OS
- ~ DSM ~ distributed shared memory

③ platforms:

- KV stores
- pub-sub/messaging
- data-processing (spark/hadoop)

* design principles:

- redundancy, replication, synchronization,
- load balancing, caching, prefetching, redirection
- ship data/app, naming/discovery

single-server web-services.

(i) CDNs

(ii) Caching

client-side caching

server-side caching

intermediate / edge
service replicas

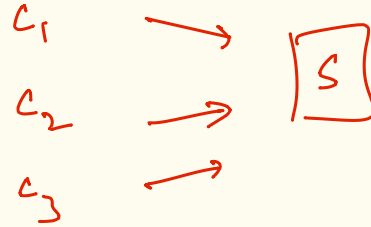
CDN nodes

http/web proxies

caching

gateway

(splitting connection)



cache coherency.

how to keep multiple
copies consistent?

(*) (i) push invalidations

(ii) refresh on access / pull-if-modified

http - HEAD

(iii) leases

last-modified:

valid for a finite duration.