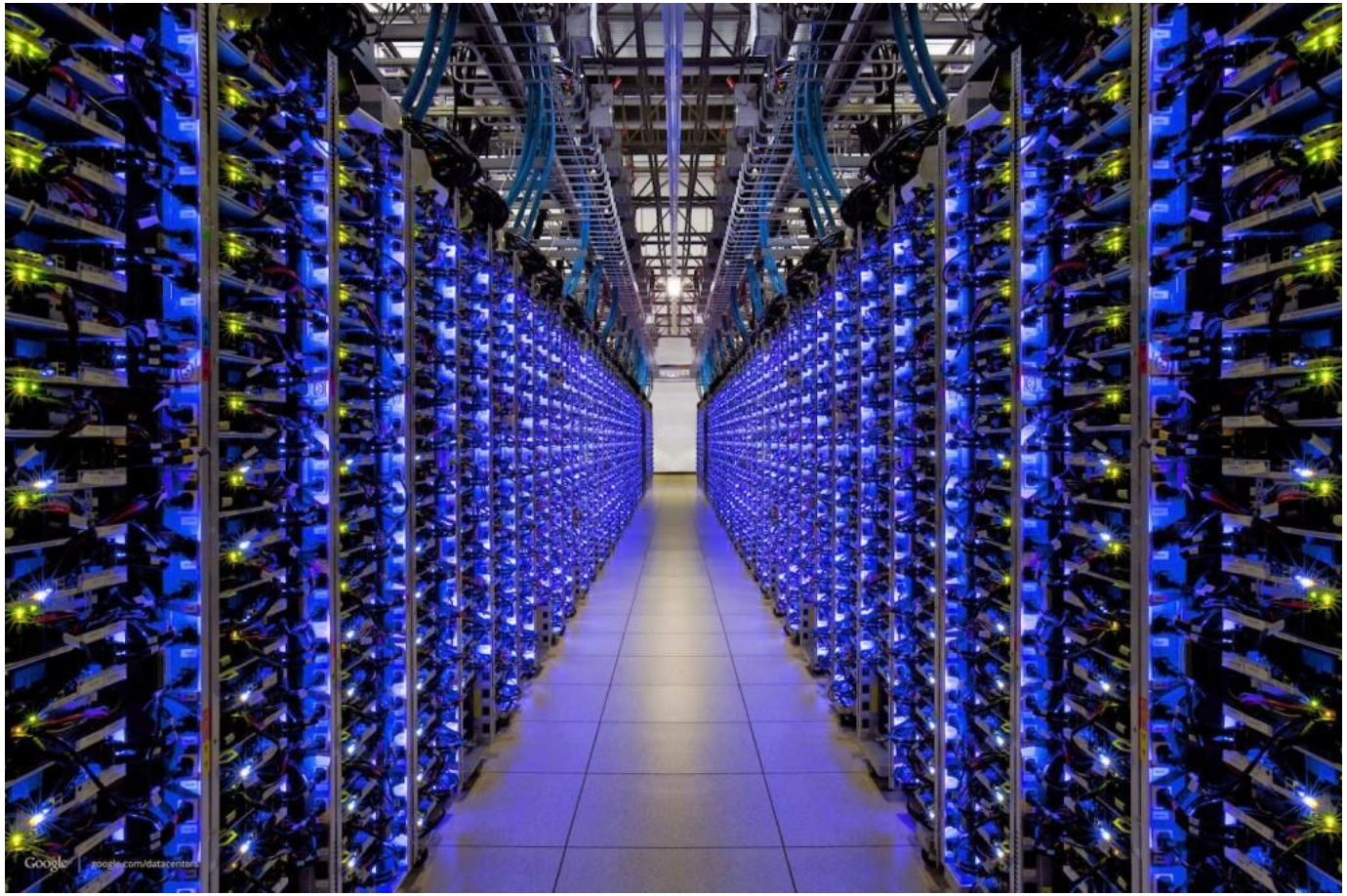
Lecture 23 – Datacenter Networks

Lecturer: Venkat Arun

Course so far

- We have covered all the main layers
- Now you largely know how the internet works
- Starting from this lecture, we will cover some special topics

What are datacenters?

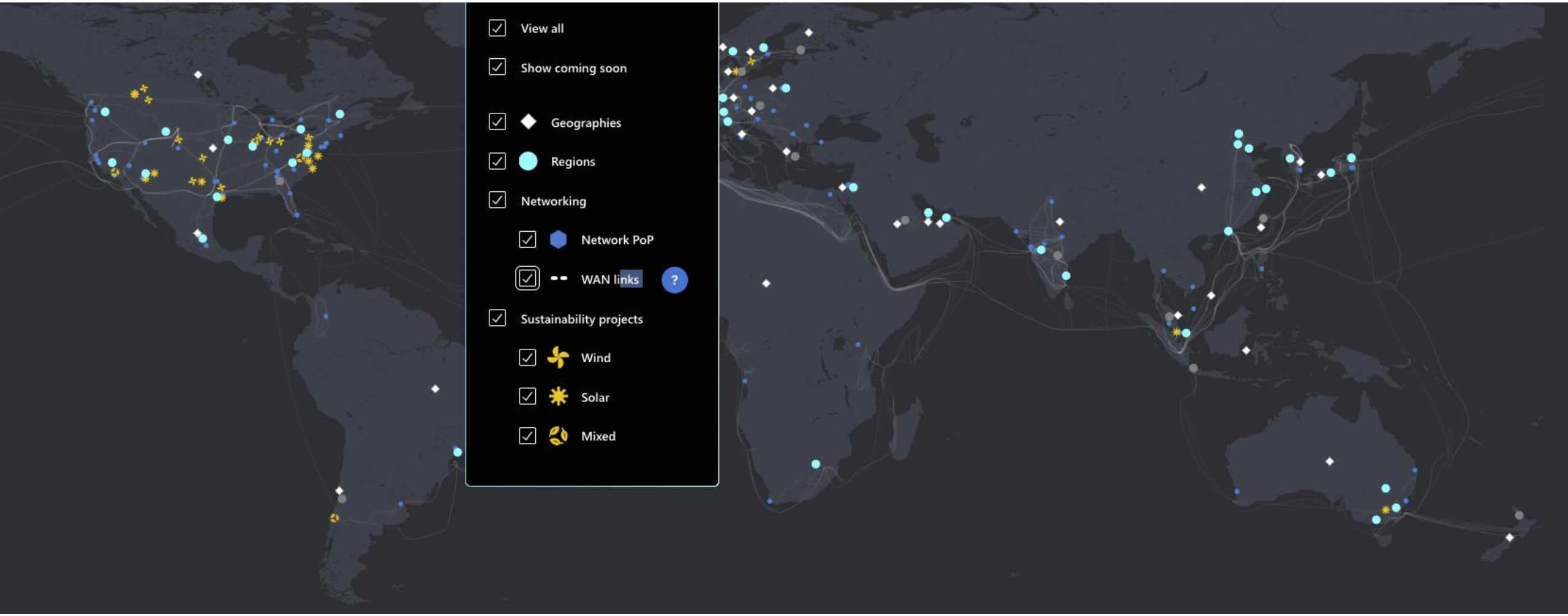


A massive warehouses of servers with reliable power, cooling and a a superfast network

Often, they are multiple warehouses of servers

This is a picture of a Microsoft datacenter complex: https://azure.microsoft.com/en-us/explore/global-infrastructure/geographies/

Some statistics: Microsoft datacenters



Some statistics: Amazon's datacenter locations

Some statistics

- One datacenter can have over 50,000 servers
- Amazon spent $35 billion on its largest datacenter site in N Virginia (https://www.datacenterfrontier.com/cloud/article/11427911/aws-has-spent-35-billion-on-its-northern-virginia-data-centers)
- Globally, datacenters have been estimated to use 1-2% of the total energy consumption of the world
- Locations are chosen based on energy availability, proximity to internet “Points of Presence (POPs)” and industrial requirements like power and water. Sometimes, they even prefer colder temperatures to offset cooling costs

Public and Private Clouds

Note: When people talk about “cloud computing” they usually mean that the servers are in a datacenter

- Organizations that need cloud computing (e.g. UT Austin, Visa, and airlines) have two options:
  - Build their own datacenter (private cloud)
  - Rent machines in someone else’s datacenter (public cloud).
    - Today, most public clouds are maintained by large organizations
    - Amazon, Microsoft and Google have the biggest market shares
    - Smaller public clouds also exist for various niches
    - CloudLab is an example of a public cloud

Benefits of a public cloud

- Everyone does not need to maintain their own physical infrastructure that ensures reliable power, network connectivity and maintenance in case some machines break
  - In large datacenters, there are so many servers that some server is always breaking down
  - For small organizations (like startups), a public cloud offers a low-cost way to get started. Maintaining infrastructure is expensive
- Organizations only need to rent as many machines as they need. This can change with time (e.g. video traffic peaks around 8PM), so they only pay for what they use
- Since public clouds have such a large number of servers, from the point of view of most non-massive organizations, they effectively have an infinite number of machines to rent

Benefits of a private cloud

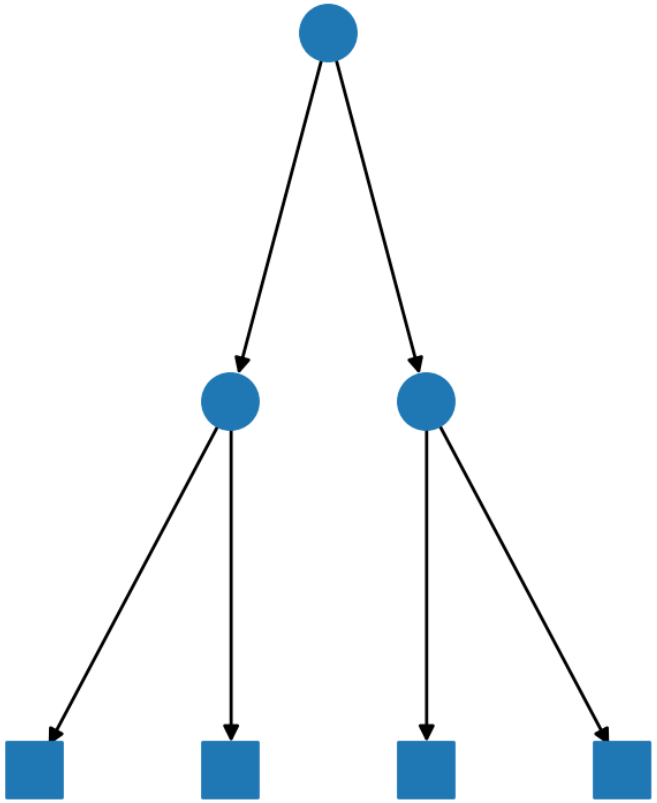
- Organizations control their own hardware. This helps if they have very specialized needs
- Some privacy compliance laws place restrictions on how data can be shared. This may require organizations like banks to own their own compute infrastructure
- Since there are only three large players in the market today, public cloud can sometimes be expensive to use

Datacenter network topology

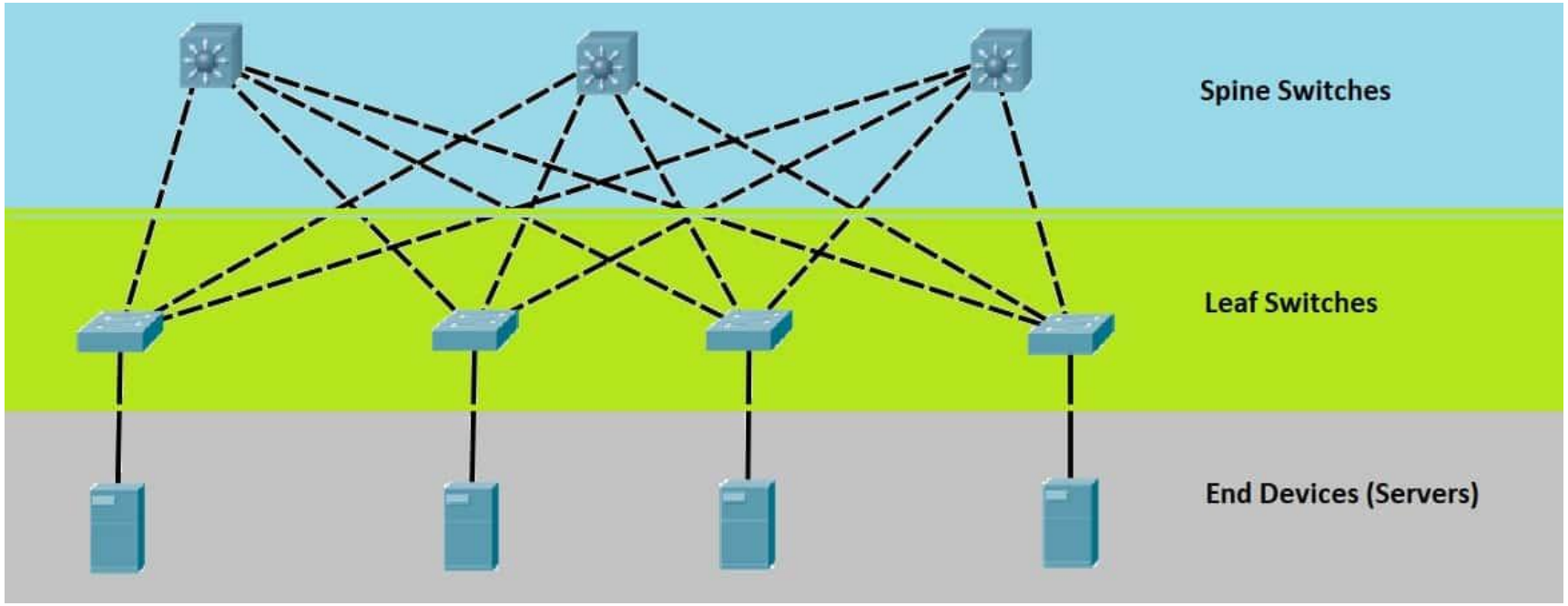
- Unlike the public internet, the fact that all the servers are in the same room and can be physically arranged as per convenience means that datacenter topologies are much more uniform
- Wide area networks tend to be less uniform because the computers' locations are determined independently of the internet (e.g. people build apartments/buildings/datacenters based on other physical constraints)
- Further, wiring is determined by where you can dig the ground to lay cables. Influenced by roads, pipelines, etc

Tree Topology

- The bottom layer has servers and the inner layers have routers
- Advantage: simple
- Disadvantages:
  - Routers higher up in the hierarchy need to be faster
  - There is only one path between any two nodes. This creates two problems:
    - If something in that path fails, the two nodes cannot communicate
    - If a link in that path is congested, the two nodes cannot communicate *fast*



Fat tree topology (aka Clos topology, leaf-spine topology)

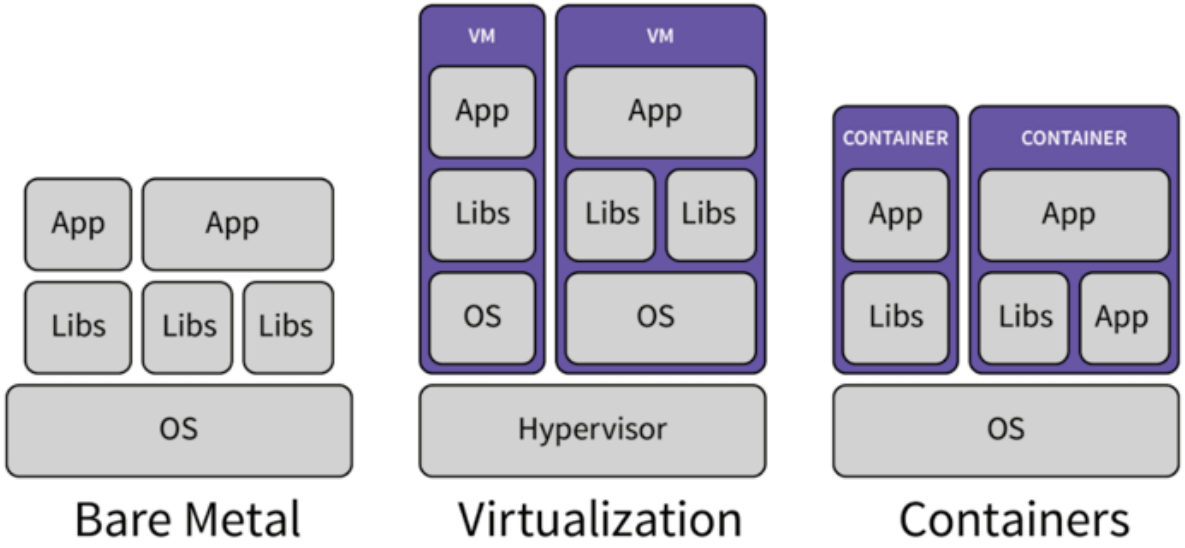


- Advantages:
  - Each router can have the same number of ports and operate at the same speed. Thus they can all be identical. Scale is achieved by adding more routers
  - There are multiple paths between any two servers. This means if one path is broken/congested, servers can pick a different path
- Internet is connected to the spine
- Most datacenter network designs today use some version of a fat tree topology

Other things a datacenter network must do

- Security:
  - Not all servers need to talk to the internet. Firewall rules prevent internet packets from reaching such servers
  - In a public cloud, servers rented by different organizations rarely need to talk to each other. Firewall rules prevent such communication unless explicitly permitted
- Performance isolation
  - One user overusing a link should not affect another user too much if they are both paying the same amount
- Addressing
  - Public clouds must make it easy for servers to find each other's IP addresses
  - Sometimes, virtual machines move from one physical server to another. Ideally, this should not change its IP address

Virtual Machines/Containers



- You rarely rent physical servers in a datacenter since they are often too big (64/128 cores) for most needs. They also make management difficult for the datacenter operator (e.g. to implement security/performance isolation)
- Instead, you rent a Virtual Machine (VM) or a container that tricks your software into thinking it is running on a (often smaller) physical machine, but in fact fits multiple VMs into the same physical machine

Microservice Applications: How do you create an application for the datacenter

Most of the content we have discussed thus far in this course are “timeless”. They have been around for decades and it is likely that they will be around for a long time in the future.

Opinion: Microservices are an exception. They became a fad ~10 years ago, and are likely to die out soon. However, they are so common today, that we should discuss them. Plus, so much software has been written this way that it will take a long time to get phased out

Also, this architecture will always be there, but in a less extreme fashion

Example Microservice application

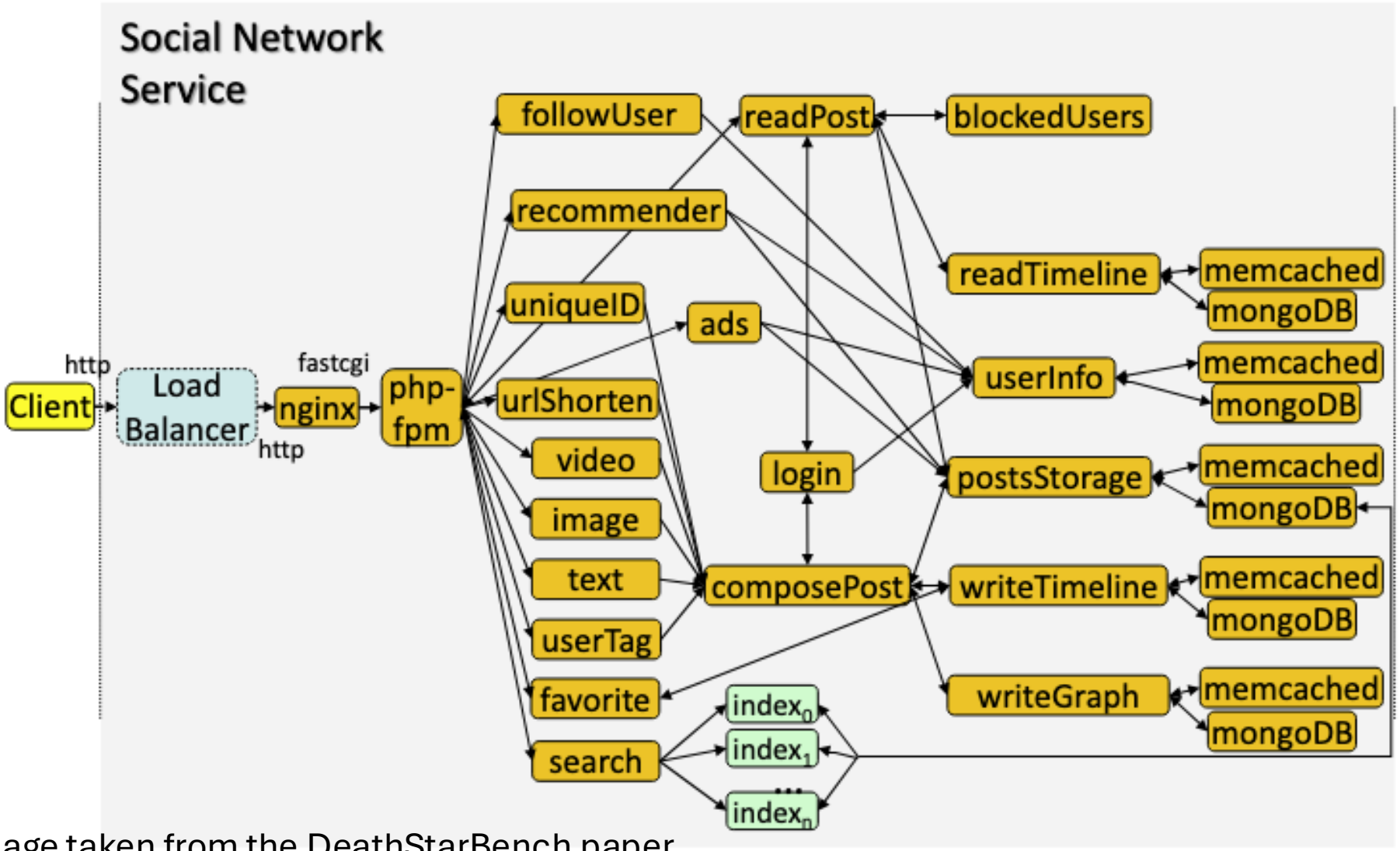


Image taken from the DeathStarBench paper

Example Microservice application

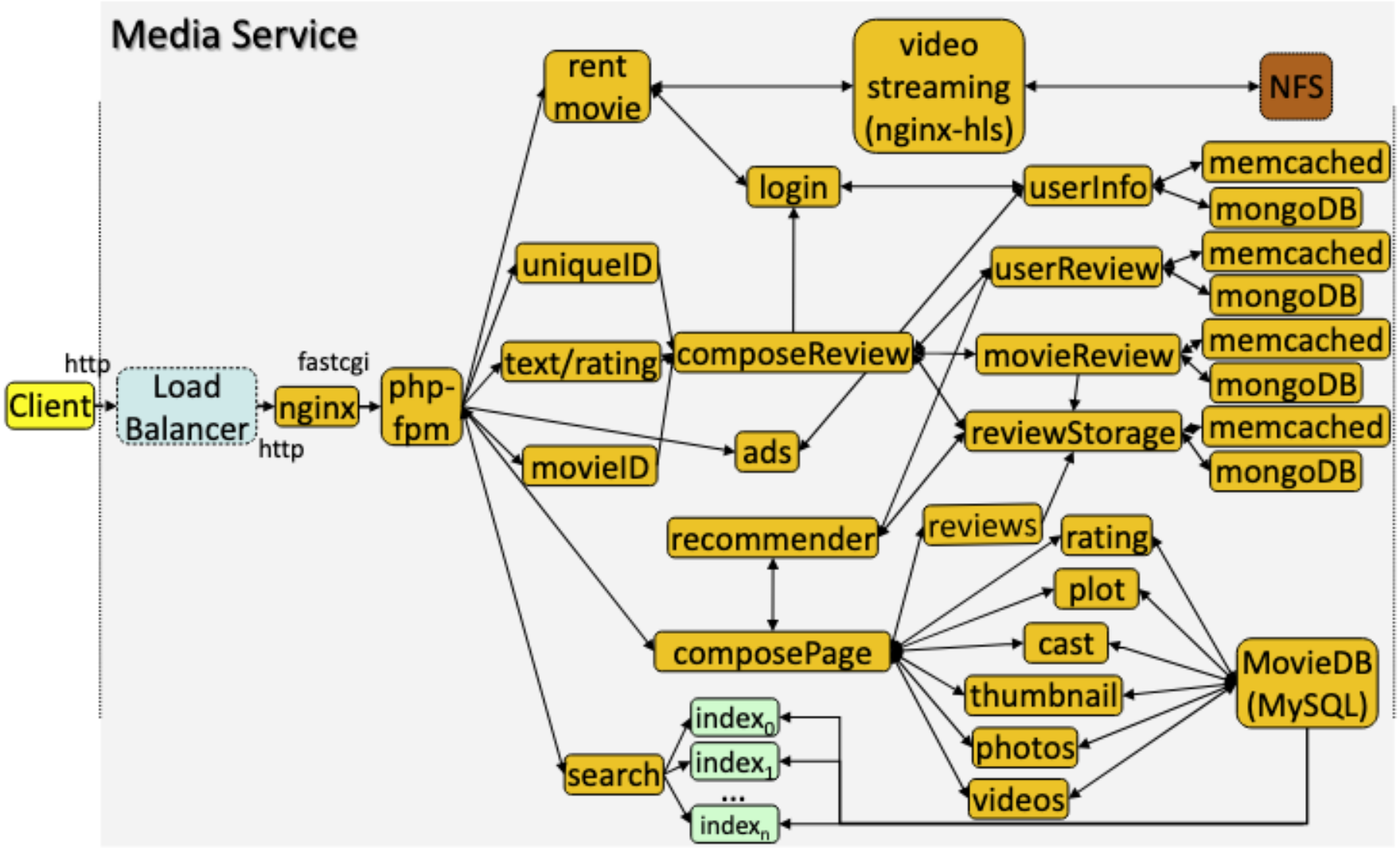


Image taken from the DeathStarBench paper