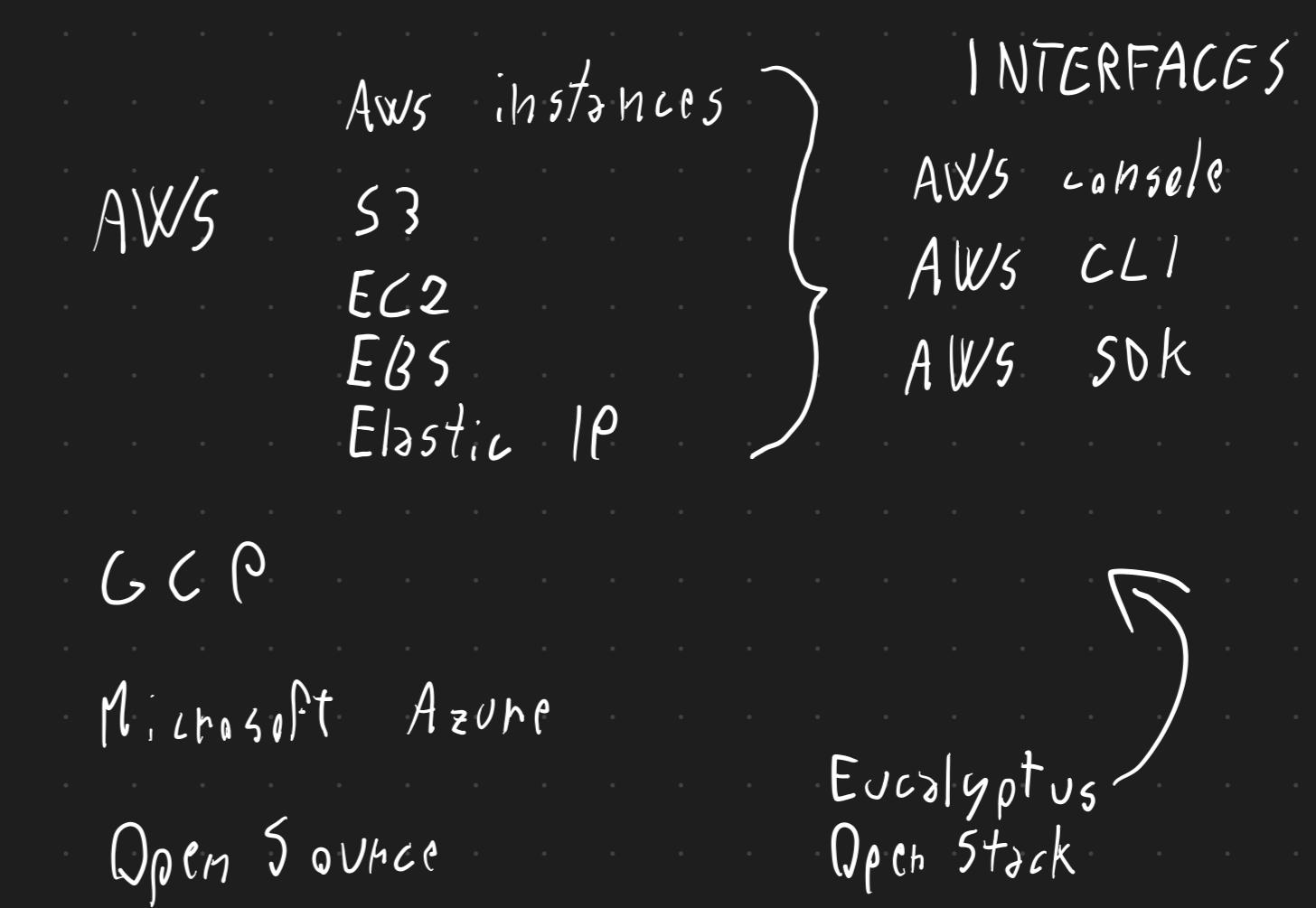


# CAPITOLO

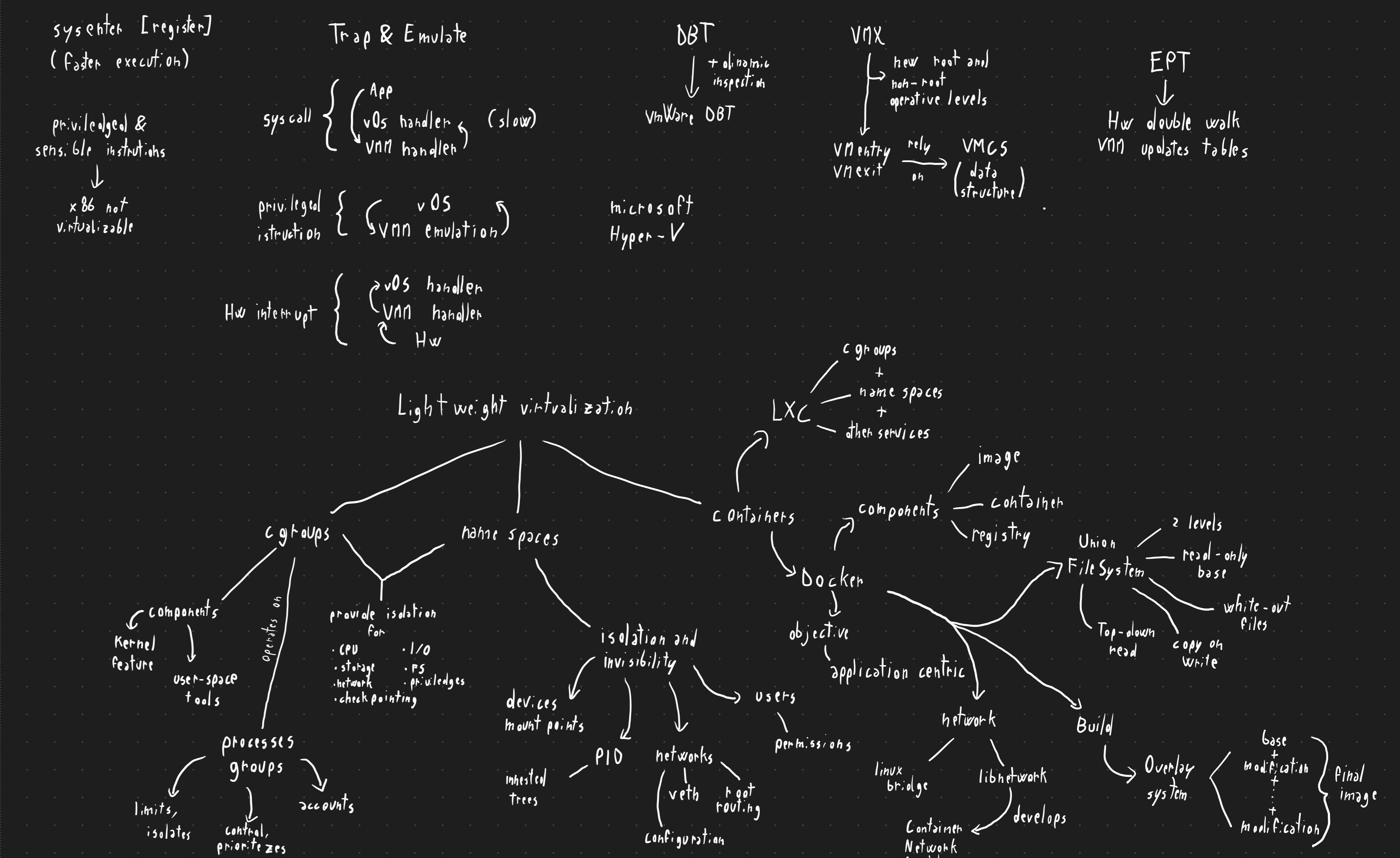
# Cloud ecosystem

Elasticity  
Deployment model

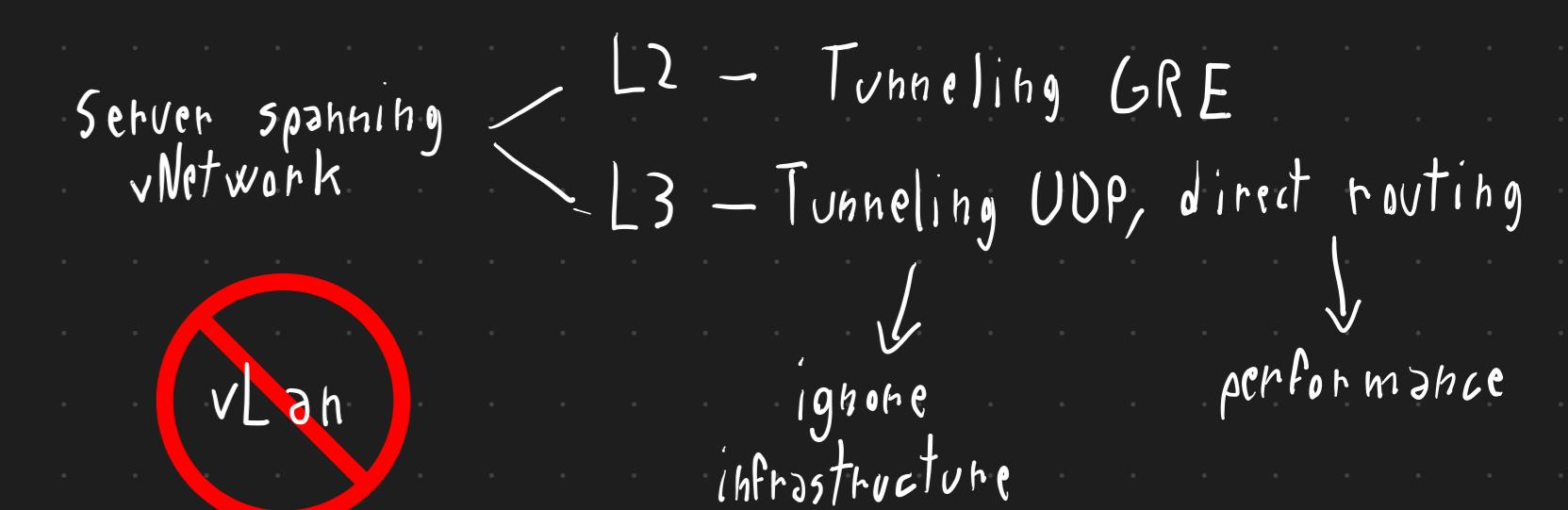
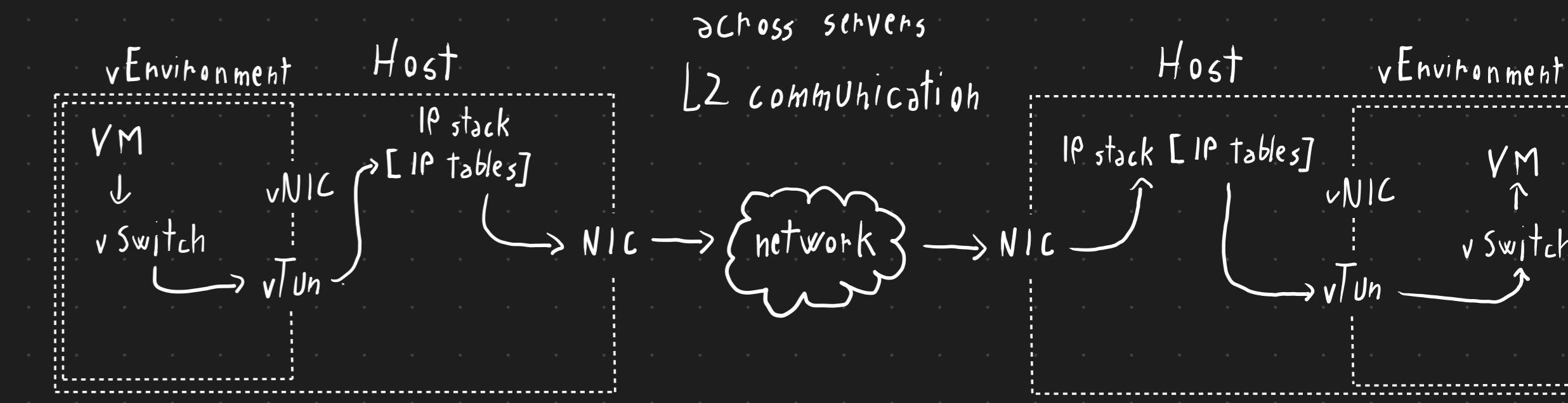
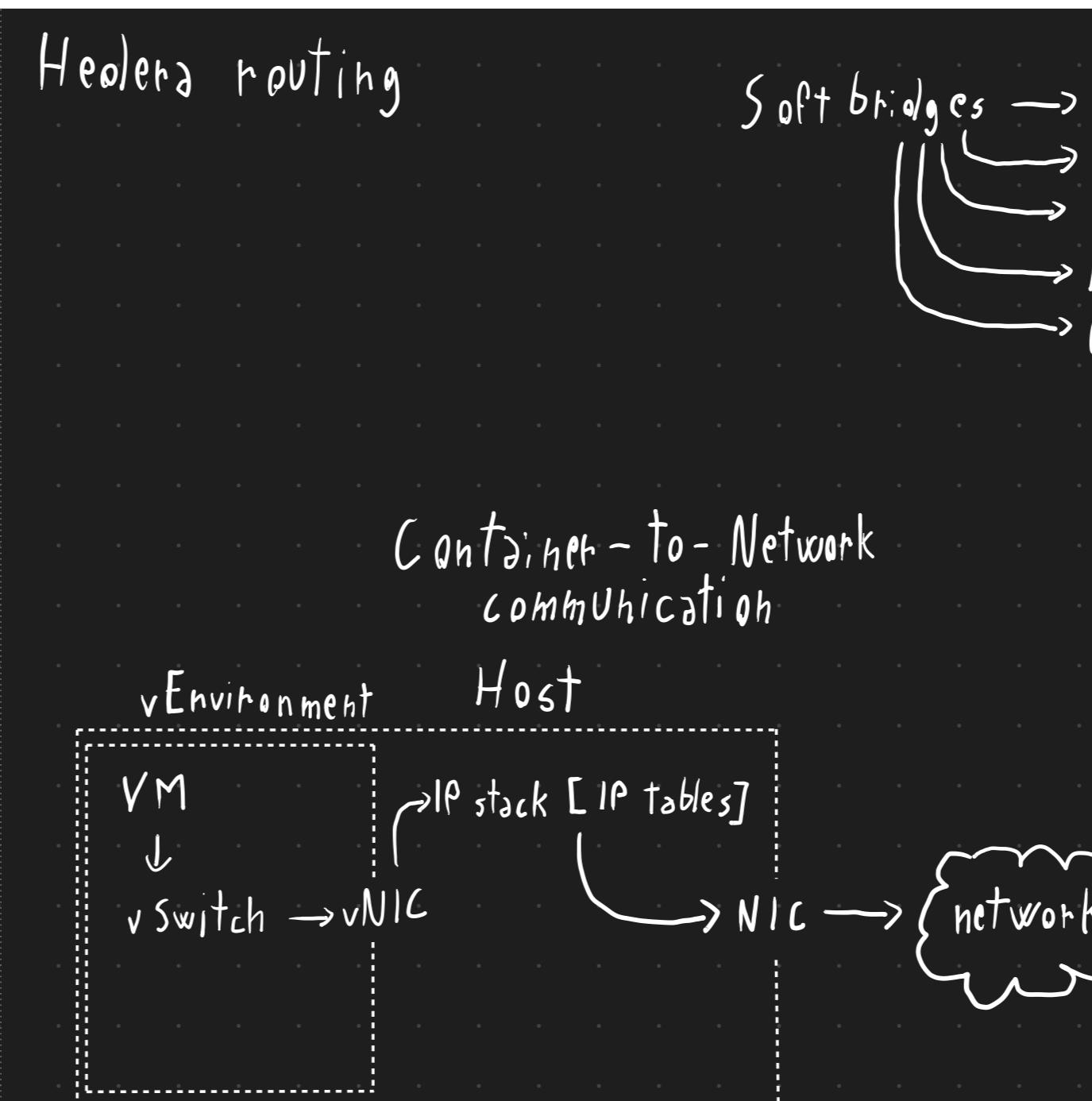
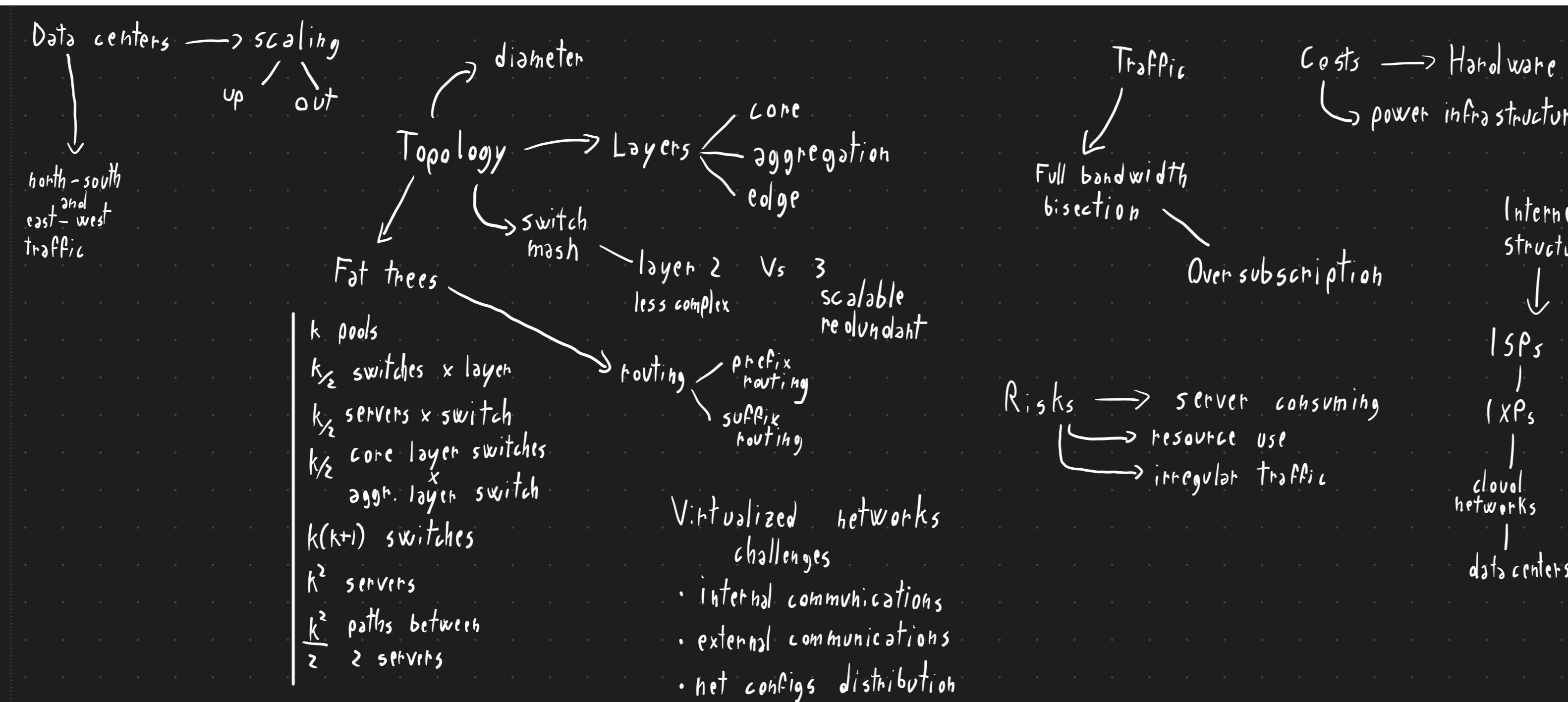
Delivery models: Haas, IaaS, PaaS, SaaS



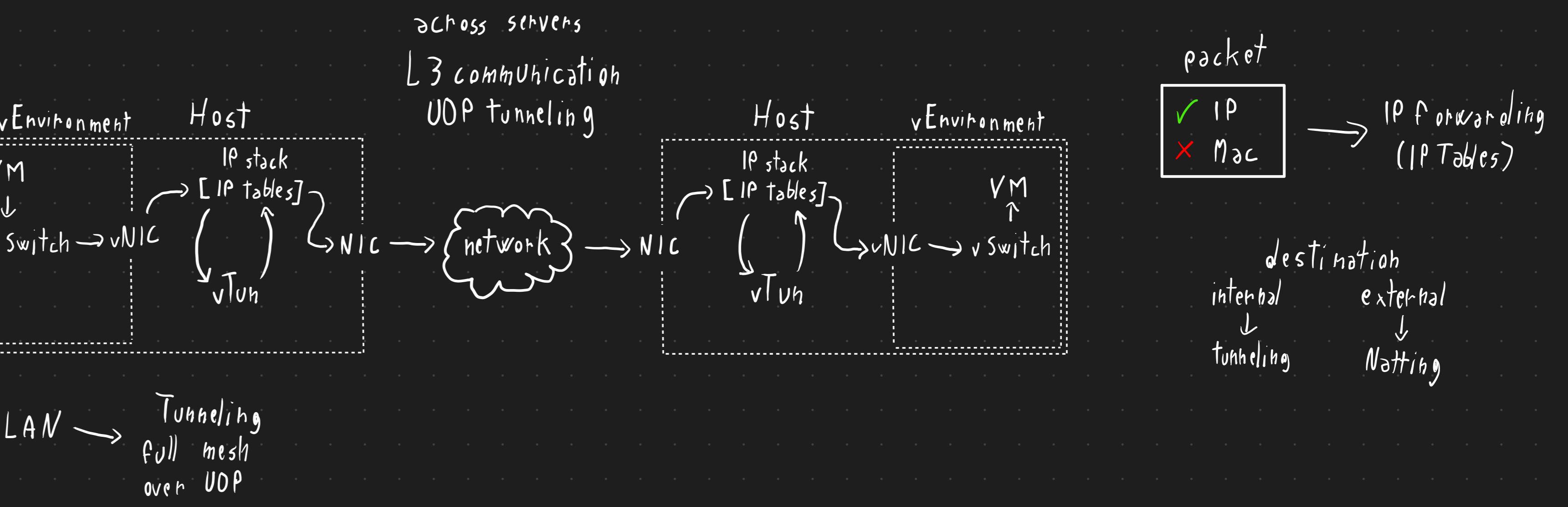
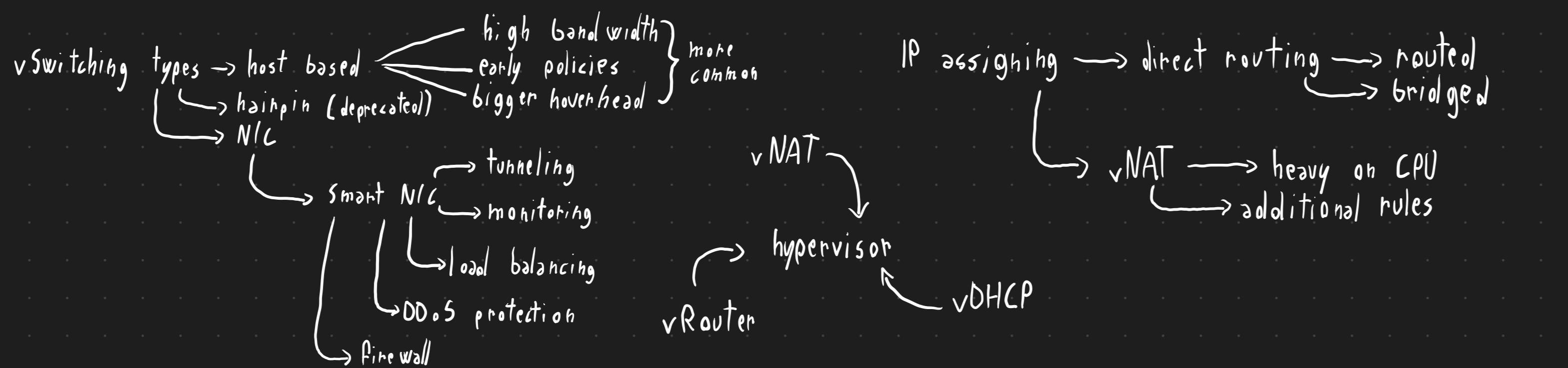
# Cloud virtualization



# Cloud Networking



# Cloud Networking



# Cloud storage

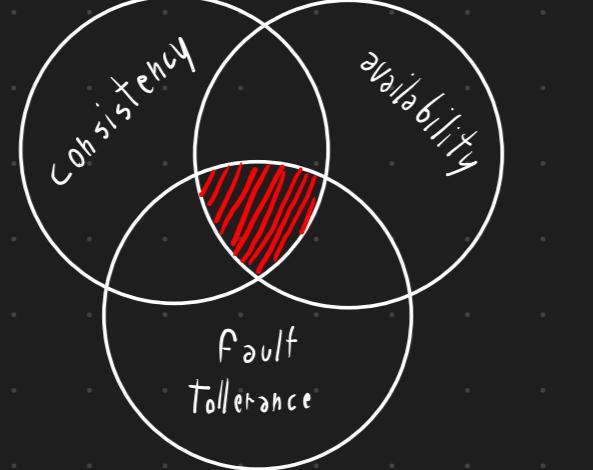
challenges → keeping costs low  
 efficient storing and data recovery  
 always available  
 concurrent access

storage area network

Distributed file systems → fault resistance  
 load balancing  
 rapid and concurrent access  
 distribution across heterogeneous platforms  
 additional → manage cache  
 resolve conflicts

Atomicity Consistency Isolation Durability → distributed databases  
 Basically read/write even available in degraded state  
 Available → asynchronous data for a short time  
 Soft state → asynchronous data for a short time  
 Eventually consistent → eventually consistent

CAP Theorem



Philosophies → reliability for the lowest cost  
 access time reduction  
 stream support  
 data multiplication for concurrent access  
 data loss risk reduction

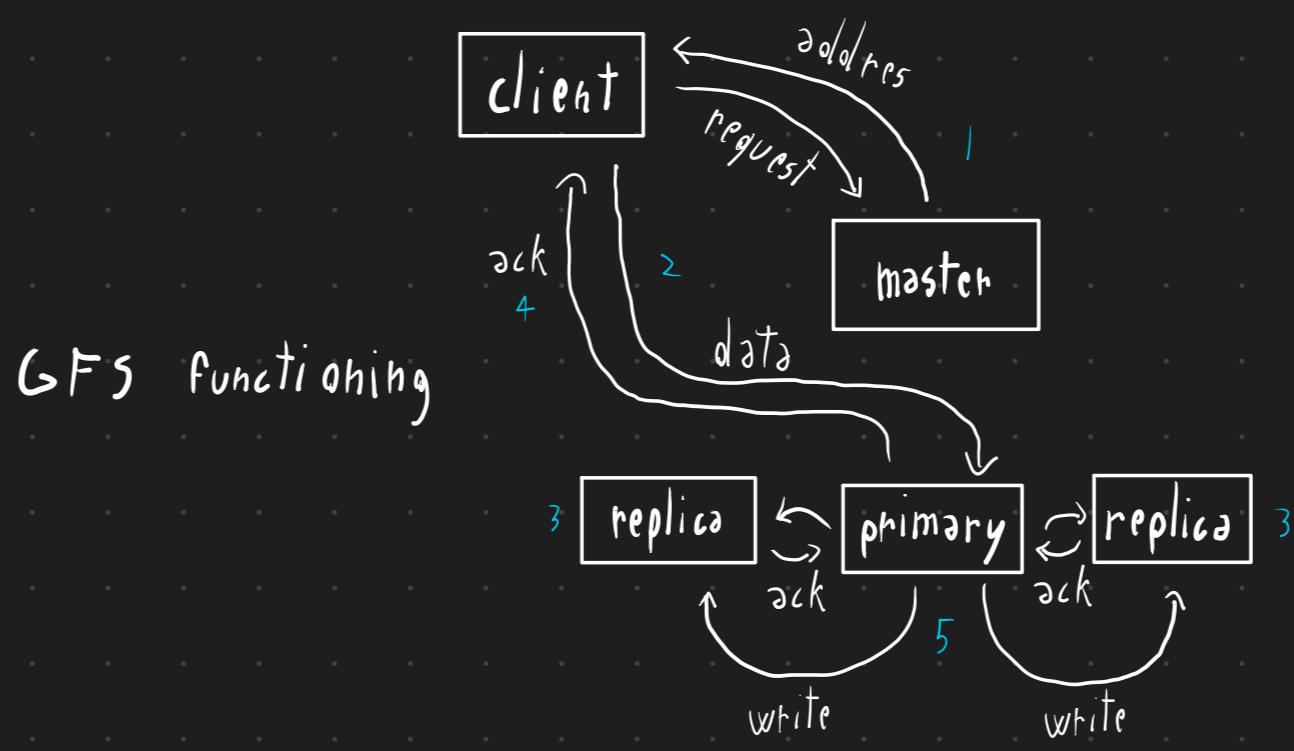
storage types → block, file, object  
 views → physical, logical  
 access mechanisms → file system, components  
 benefits → efficient data distribution, better access and API  
 components → data, meta data, unique identifier

Locks → consensus  
 bring unstable systems to a stale state  
 access permissions → locks  
 mechanism to reach a shared choice

Unix File System → inode  
 file descriptors, abstraction layers, path, inode, block, ...  
 Network File System → client-server paradigm, vNode, NFS  
 Remote Procedure Calls, File Handlers

General Parallel FS → data striping, write ahead log file

Google FS → mass processing for answer time  
 64 MB chunks with at least 3 replicas in different servers  
 master manages consistency ops, critical ops, data delivery topology



Google Big Table → CAP → consistency, fault tolerance  
 distributed, ordered, multi-directional map

organization → global id  
 family grouped columns  
 version ordered cells

GBT servers → Master server → assigns tablets  
 makes data available to the client  
 divides tablets  
 Tablet servers → manages tablets and their health  
 ingests tablets  
 load balances

GBT Tablet → table section  
 control string table format → index block

organization → data tablets → points to meta data tablets → points to root tablet → points to stores

GBT request

