Carnegie Mellon

institute for SOFTWARE RESEARCH

Applied Distributed Systems

The Cloud 1

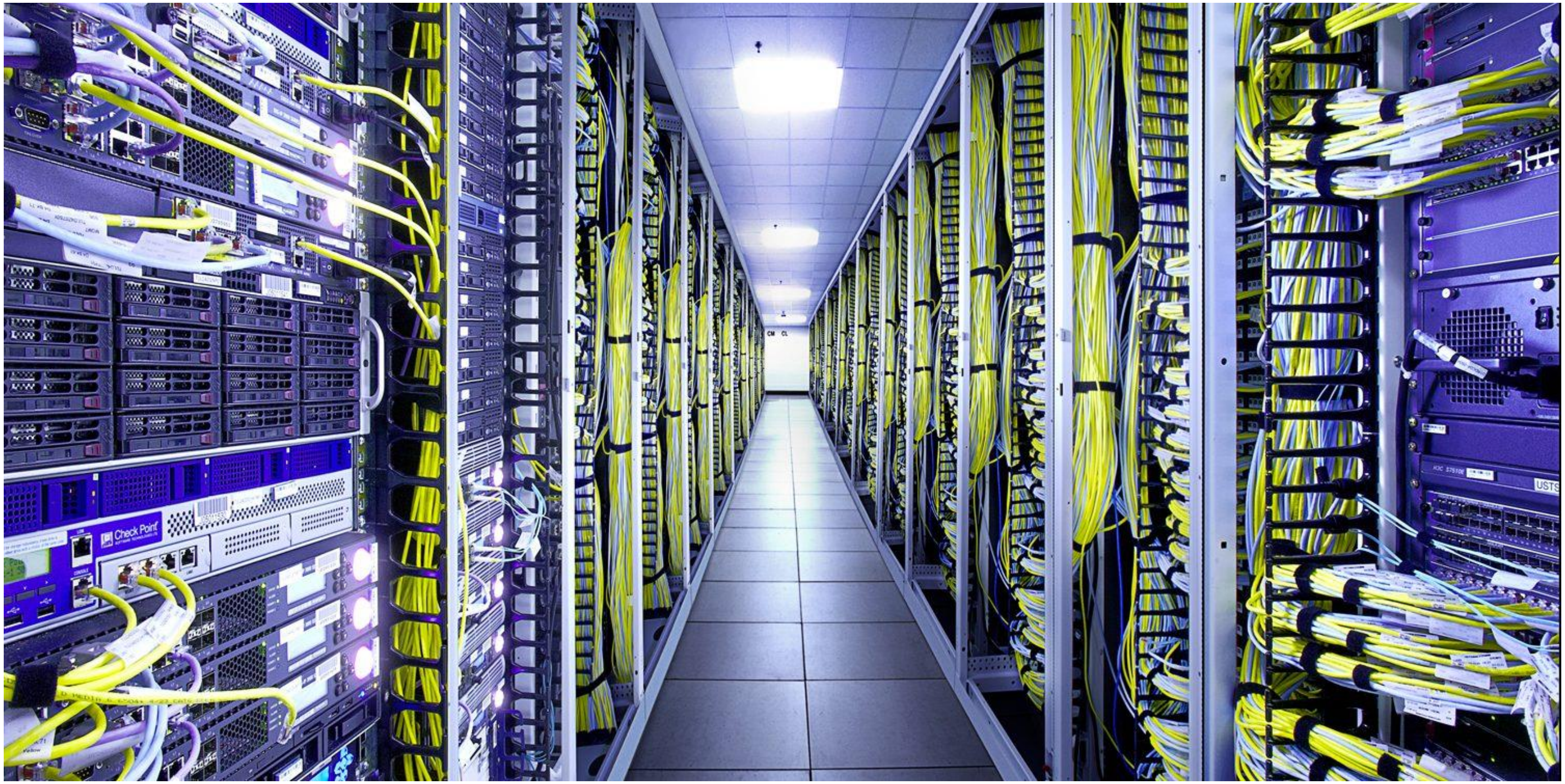
Overview

- **Structure**
- Scaling Service Capacity
- Time

Data centers in the cloud

- A cloud provider (AWS, Google, Microsoft) maintains data centers around the world.
- Each data center has ~100,000 computers.
- Limited by power and cooling considerations.

Data Center



Organization of data centers

- Each data center has independent power supply, independent fire control, independent security, etc
- Data centers are collected into availability zones and availability zones are collected into geographic regions.
- AWS now has 24 geographic regions and has announced plans for five more regions in Indonesia, Japan, and Spain, India, and Switzerland.

Allocating a virtual machine in AWS - 1

- A user wishes to allocate a virtual machine in AWS
  - The user specifies
    - A region
    - Availability zone
    - Image to load into virtual machine
    - …

Allocating a virtual machine in AWS - 2

- AWS management software
  - Finds a server in that region and availability zone with spare capacity
  - Allocates a virtual machine in that server
  - Assigns IP address to that virtual machine (public)
  - Loads image into that virtual machine
- VM can then send and receive messages

AWS access stastics

- Amazon reports: we [AWS] authenticate and authorize over 400 million API calls EVERY SECOND.
- Think about what their architecture must be to support this kind of load.

Overview

- Structure
- **Scaling Service Capacity**
- Time

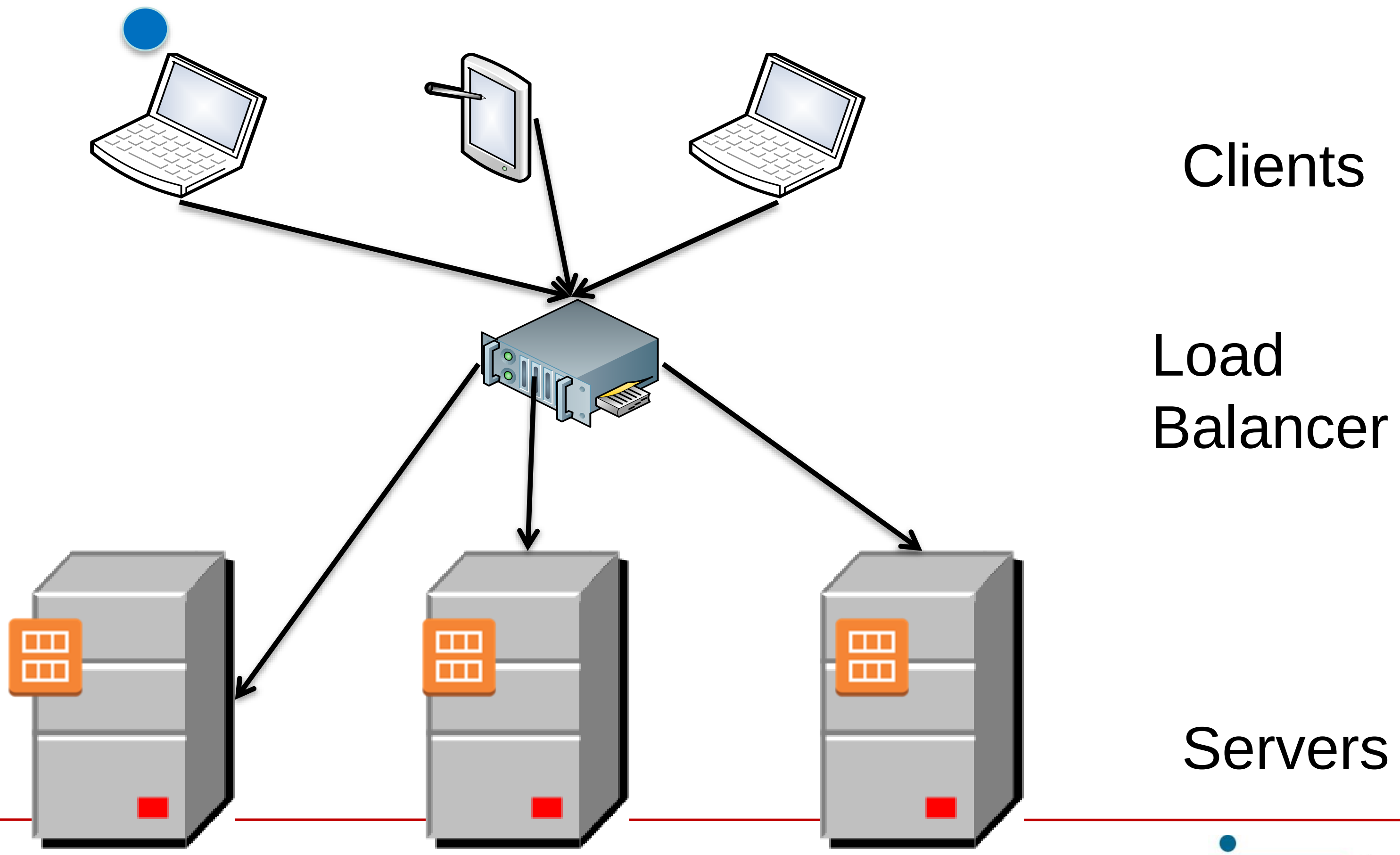
Load Balancer - 1

- One server may not suffice for all of the requests for a given service.
  - Have multiple servers supplying the same service.
  - Use “load balancer” to distribute requests.
- Server is registered with load balancer upon initialization

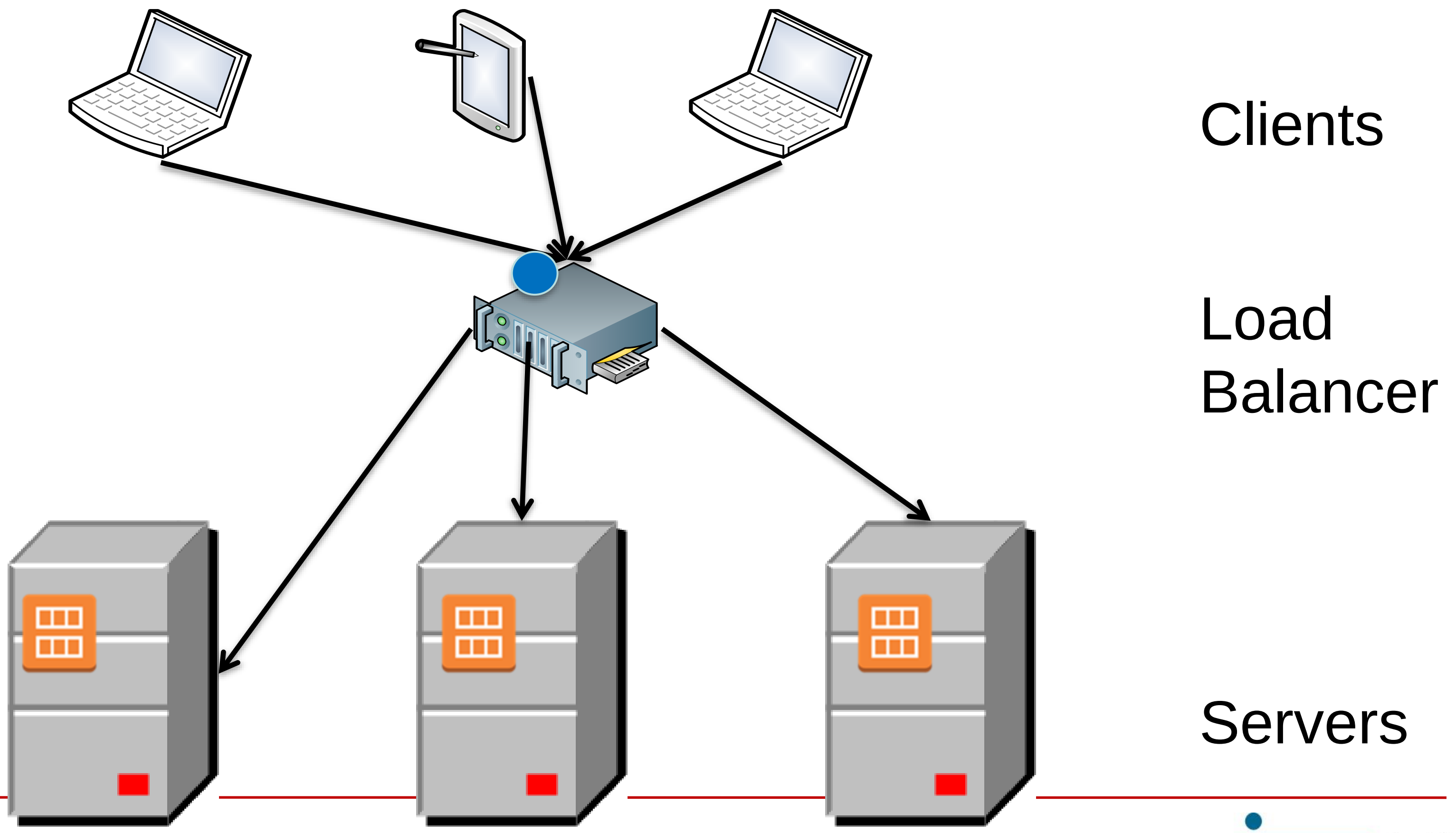
Load Balancer - 2

- Load balancer monitors health of servers and knows which ones are healthy.
- All servers managed by one load balancer are identical
- Load balancer IP is returned from DNS server when client requests URL of service.

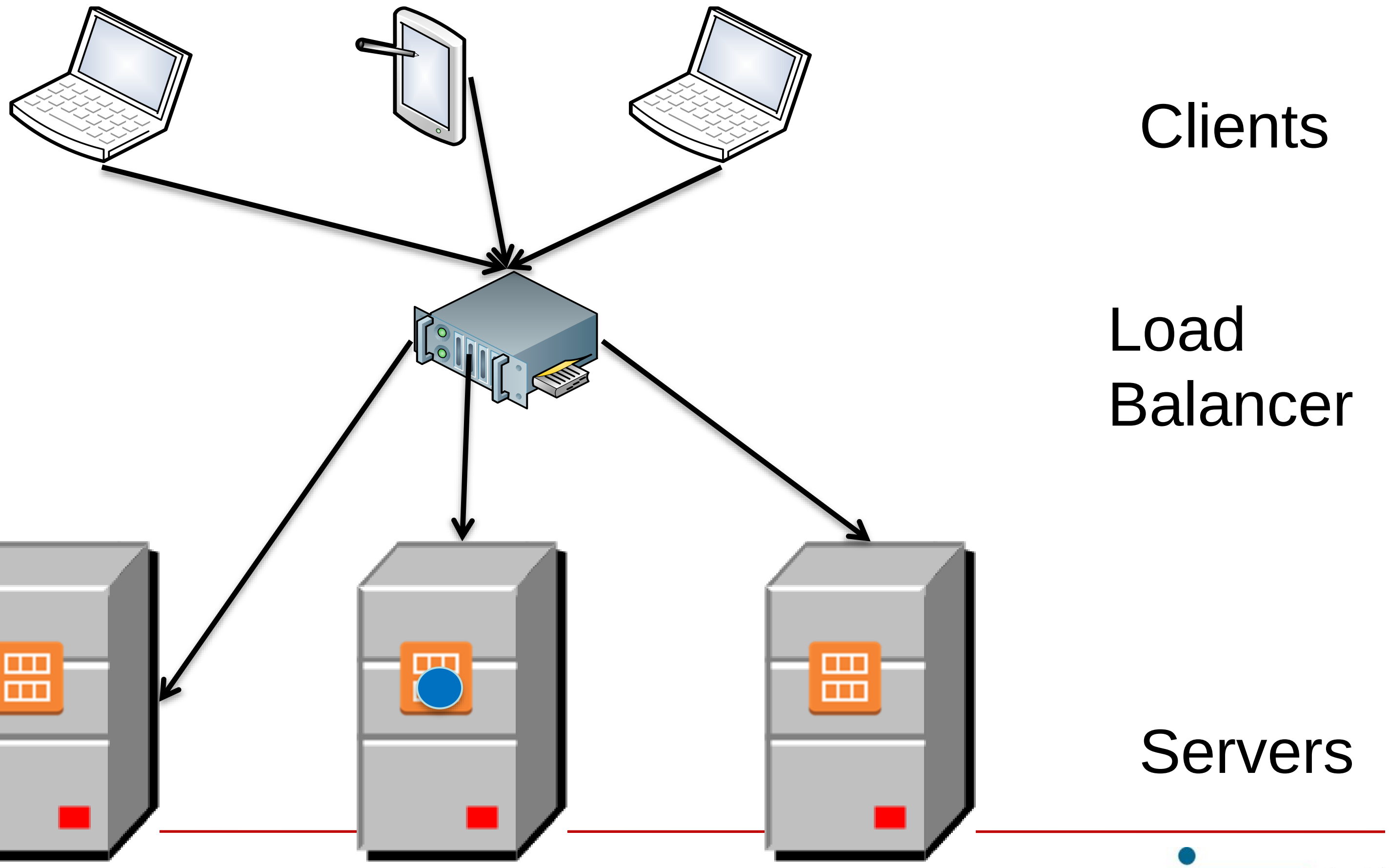
Message sequence – client makes a request



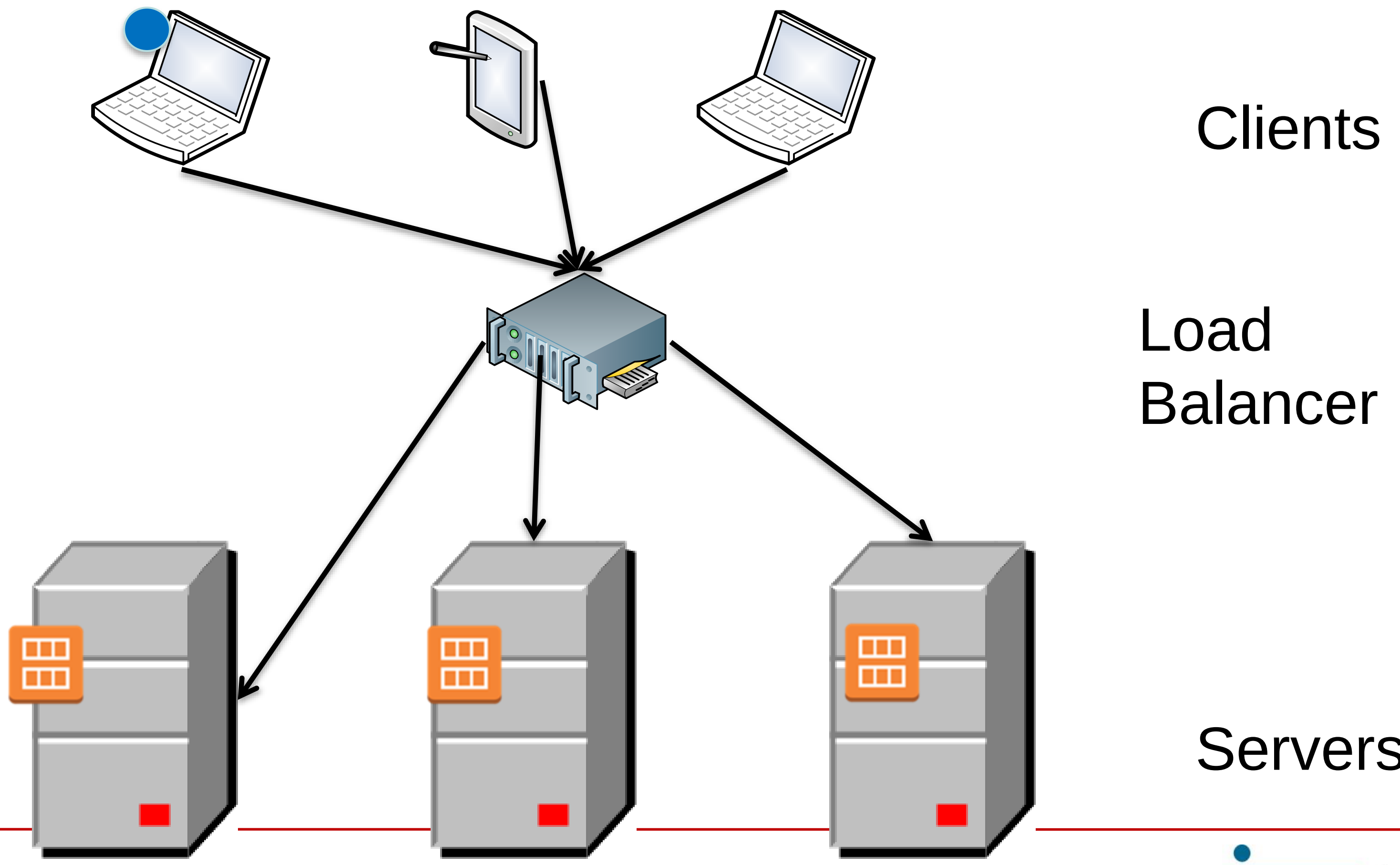
Message sequence- request arrives at load balancer



Message sequence – request is send to one server



Message sequence – reply goes directly back to sender



Note IP manipulation

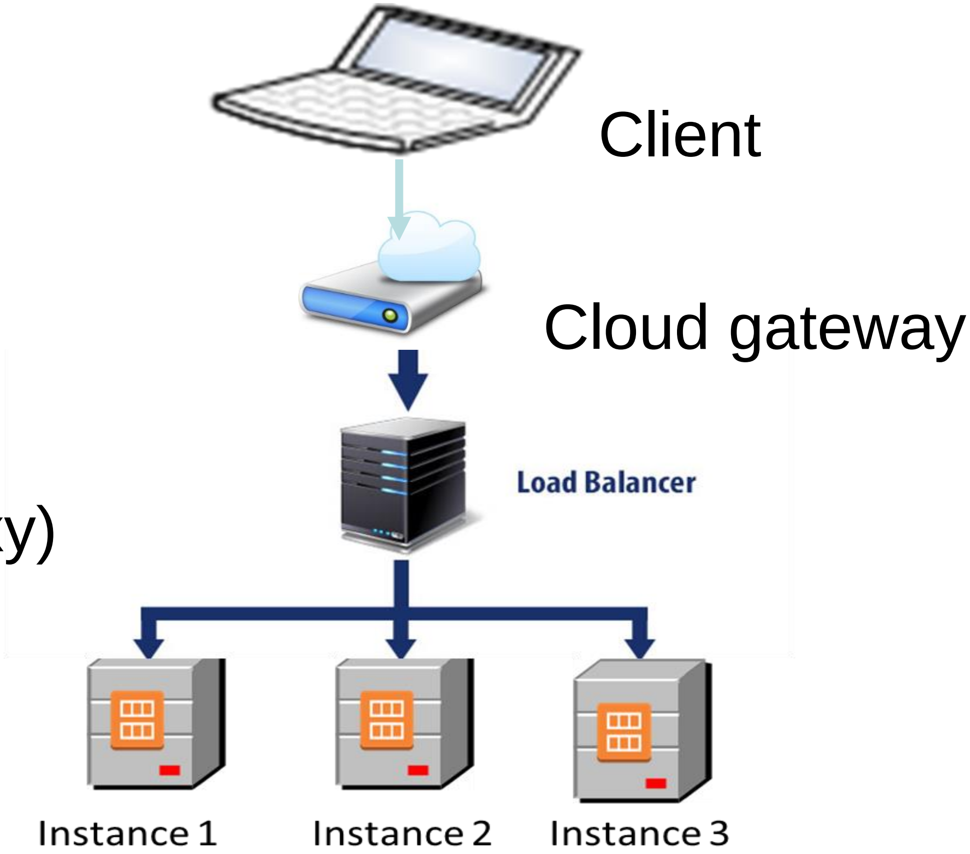
- Server always sends message back to what it thinks is the sender.
- Load balancer changes destination IP but not the source. Then reply goes directly back to client
- Load balancer (now acting as a proxy) can change origin as well. In this case, reply goes back to load balancer which must change destination (of reply) back to original client.

Routing algorithms

- Load balancers use variety of algorithms to choose instance for message
  - Round robin. Rotate requests evenly
  - Weighted round robin. Rotate requests according to some weighting.
  - Hashing – IP address of source to determine instance. Means that a request from a particular client always sent to same instance as long as it is still in service.
- Note that these algorithms do not require knowledge of an instance's load.

Combining pictures

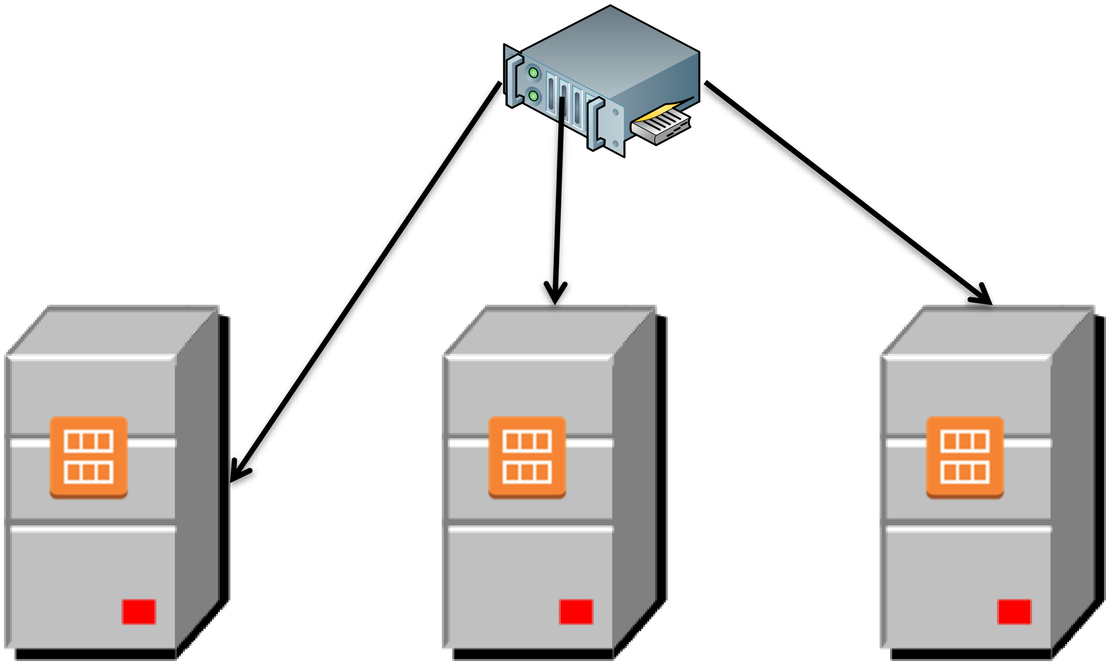
- IP payload address is modified multiple times before message gets to server
- Return message from instance to client may go through the gateway (proxy) or may go directly back to the client.



Suppose servers become overloaded

- As load grows, existing resources may not be sufficient.
- Autoscaling is a mechanism for creating new instances of a server.
- Set up a collection of rules that determine
  - Under what conditions are new servers added
  - Under what conditions are servers deleted

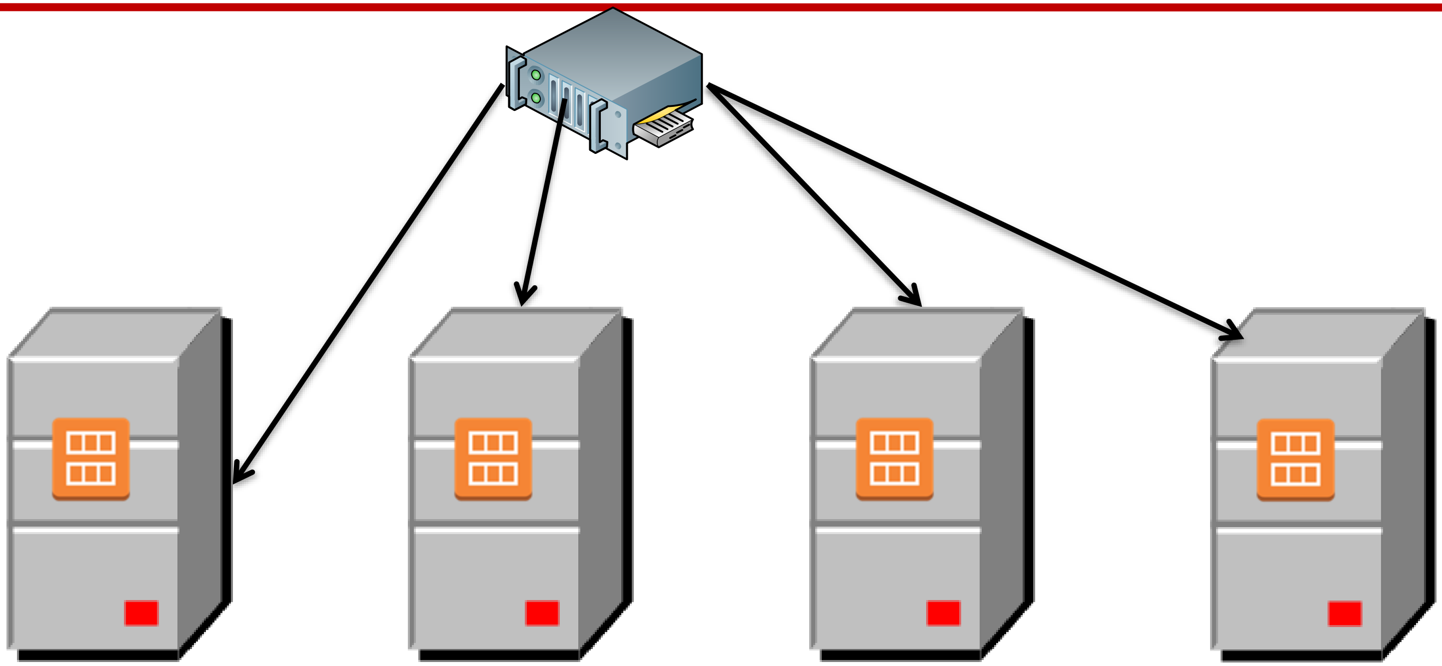
First there were three servers



Load Balancer

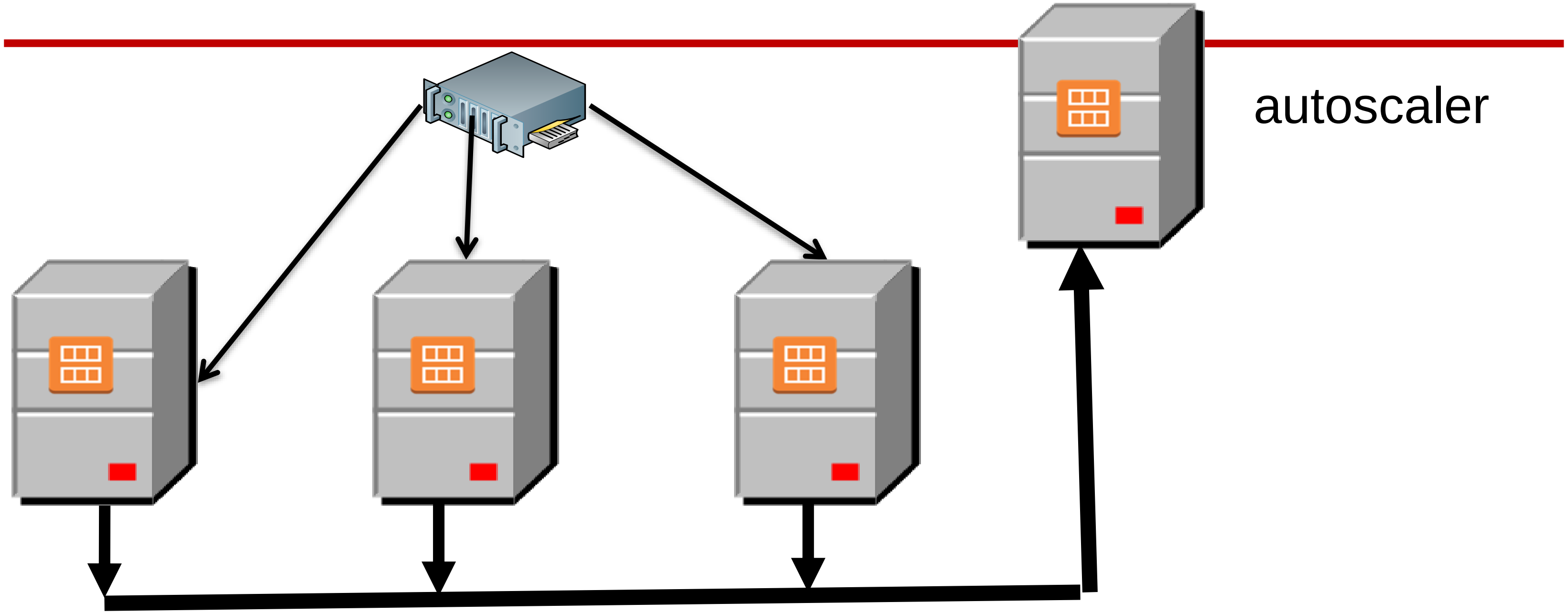
Servers

Now there are four



Issues:
- What makes the decision to add a new server?
- How does the new server get loaded with software?
- How does the load balancer know about the new server?

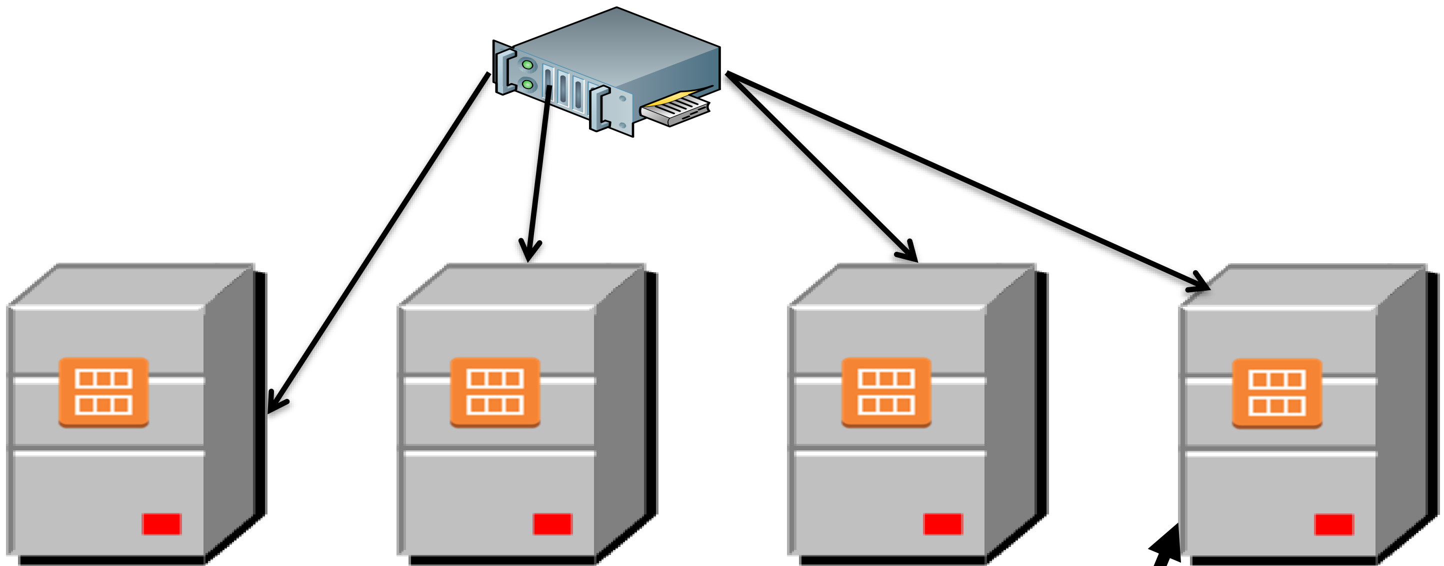
Making the decision



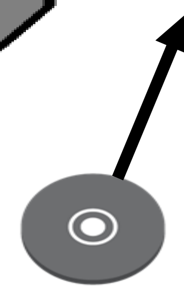
Each server reports its CPU and I/O usage to an autoscaler

Autoscaler has a collection of rules to determine whether to add new server. E.g. one server is over 80% utilization for 15 minutes

Loading the new server with software

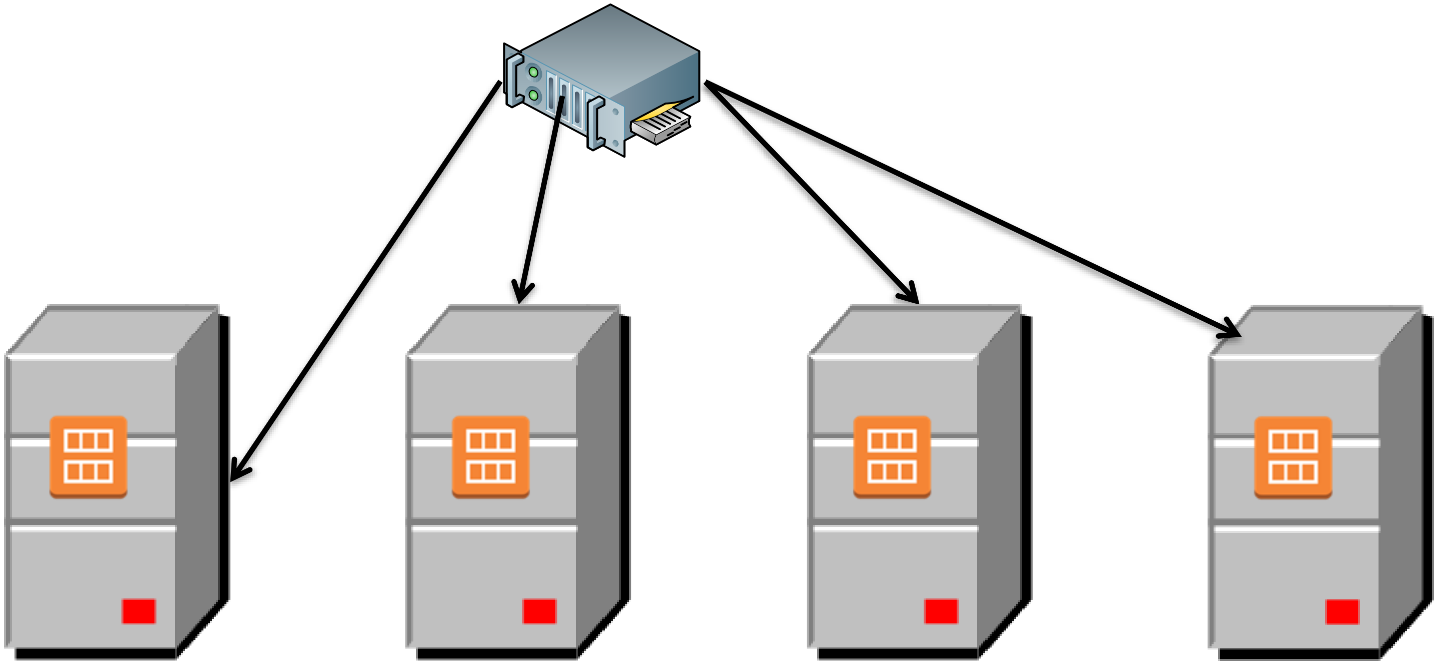


When a new instance is created, it is created based on a launch configuration, security groups, etc



Configuration specifications

Making the load balancer new server aware



The new server is registered with a load balancer by the autoscaler.

Overview

- Structure
- Scaling Service Capacity
- **Time**

Time In a distributed system

- Many protocols involve putting a time stamp on messages or logs
- Problem is that clocks drift and so clock time may be different on different computers.

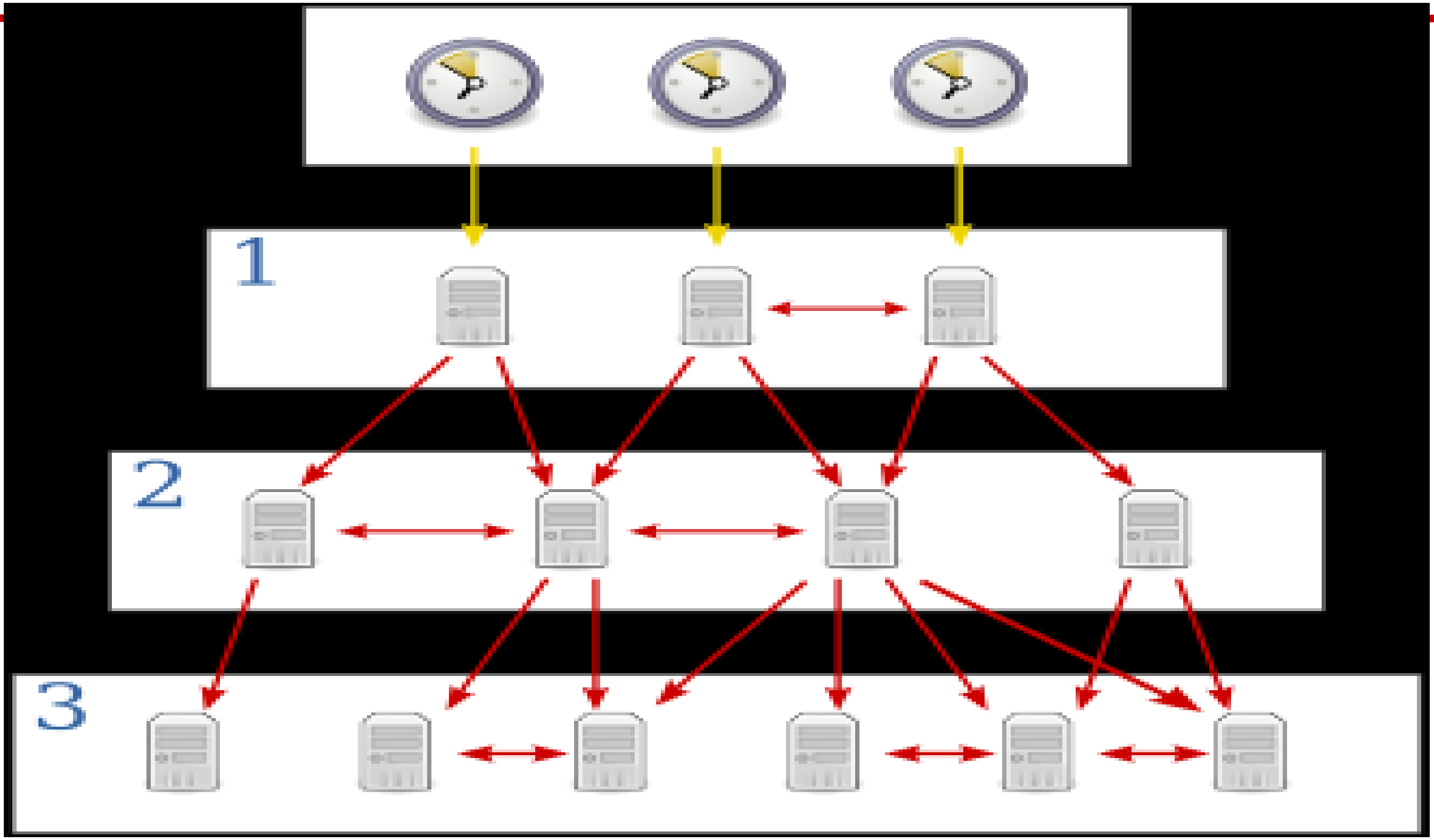
Clock drift

- Clocks on a modern processor may drift 1 second every million seconds (11+ days)
- This means two computers will have different time readings.
- Relative times within a single computer will be accurate
- But using time to sequence events across a network will not be accurate.

Synchronizing clocks across a network

- Suppose two different computers are connected via a network. How do they synchronize their clocks?
- If one computer sends its time reading to another, it takes time for the message to arrive.
- NTP (Network Time Protocol) can be used to synchronize time on a collection of computers. Involves having a server that sends out network time periodically.
  - Accurate to around 1 millisecond in local area networks
  - Accurate to around 10 milliseconds over public internet
  - Congestion can cause errors of 100 milliseconds or more.

NTP



Suppose NTP is insufficiently accurate

- Financial industry spent100s of millions of dollars to reduce latency between Chicago and New York by 3 milliseconds.
  - Well within error range of NTP
- GPS time is accurate within
  - 14 nanoseconds (theoretically)
  - 100 nanoseconds (mostly)
- Timestamp messages with GPS time
  - Used by electric companies to measure phase angle
- Atomic clocks
  - Used by Google to coordinate time across all of their distributed systems.

Tracking operations across the cloud

- Suppose a request involves multiple computers in the cloud
- Tracking this request with time involves expensive mechanisms (GPS, Atomic clock).
- Other mechanisms must be used to track requests (discussed later).

Summary

- The cloud is composed of a collection of independent data centers scattered around the globe
- Load balancers are used to distributed requests among identical servers
- Autoscaling is a mechanism to increase the number of servers if they get overloaded.
- Time within a server is accurate, across servers it is inaccurate.