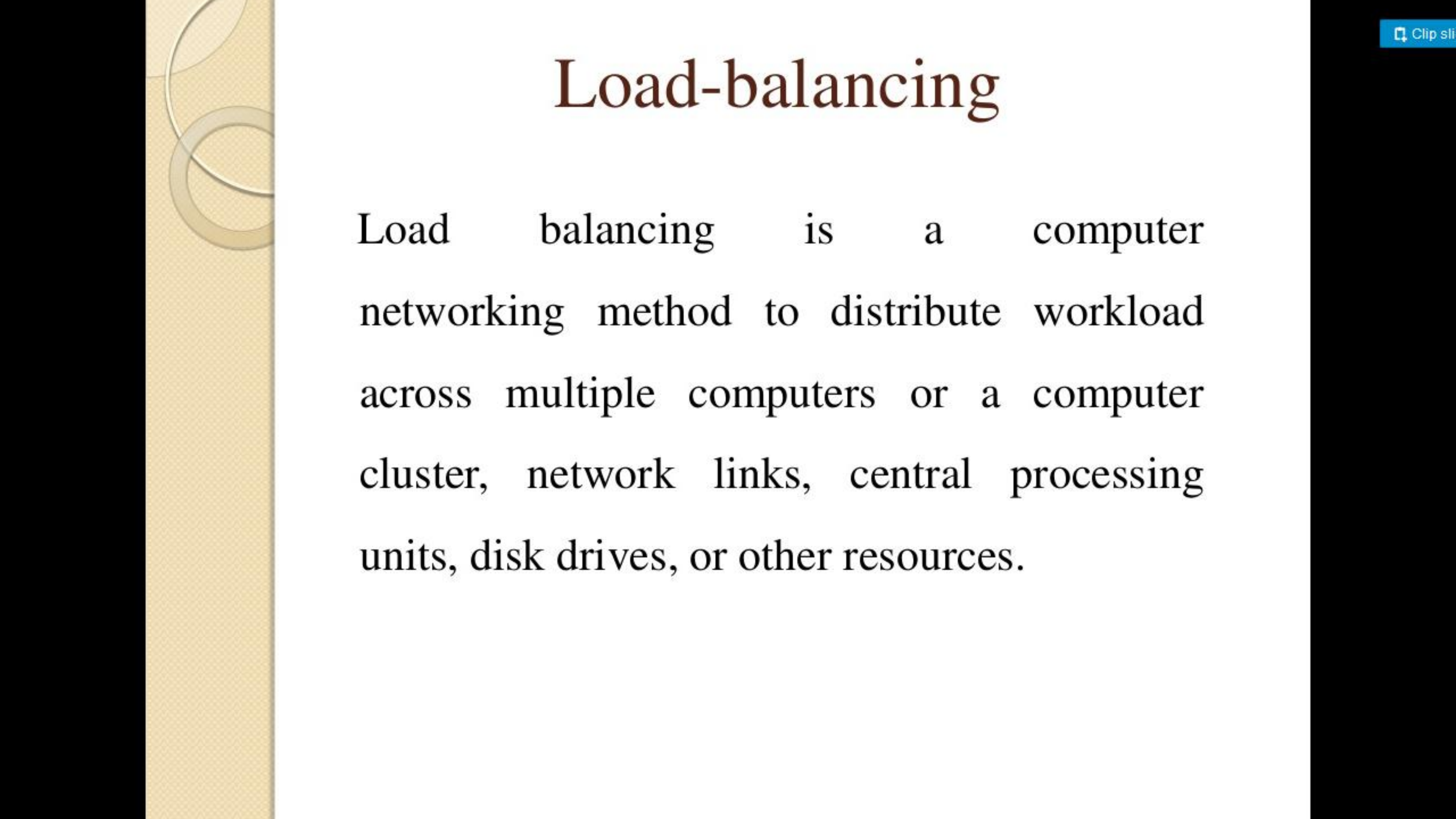
Load Balancing

Contents

- What is load Balancing ?
- Goals of Load Balancing
- Types of Load Balancing
    - Static Load Balancing
    - Dynamic Load Balancing
- Strategies of Load Balancing
    - Centralised
    - Distributed



Load-balancing

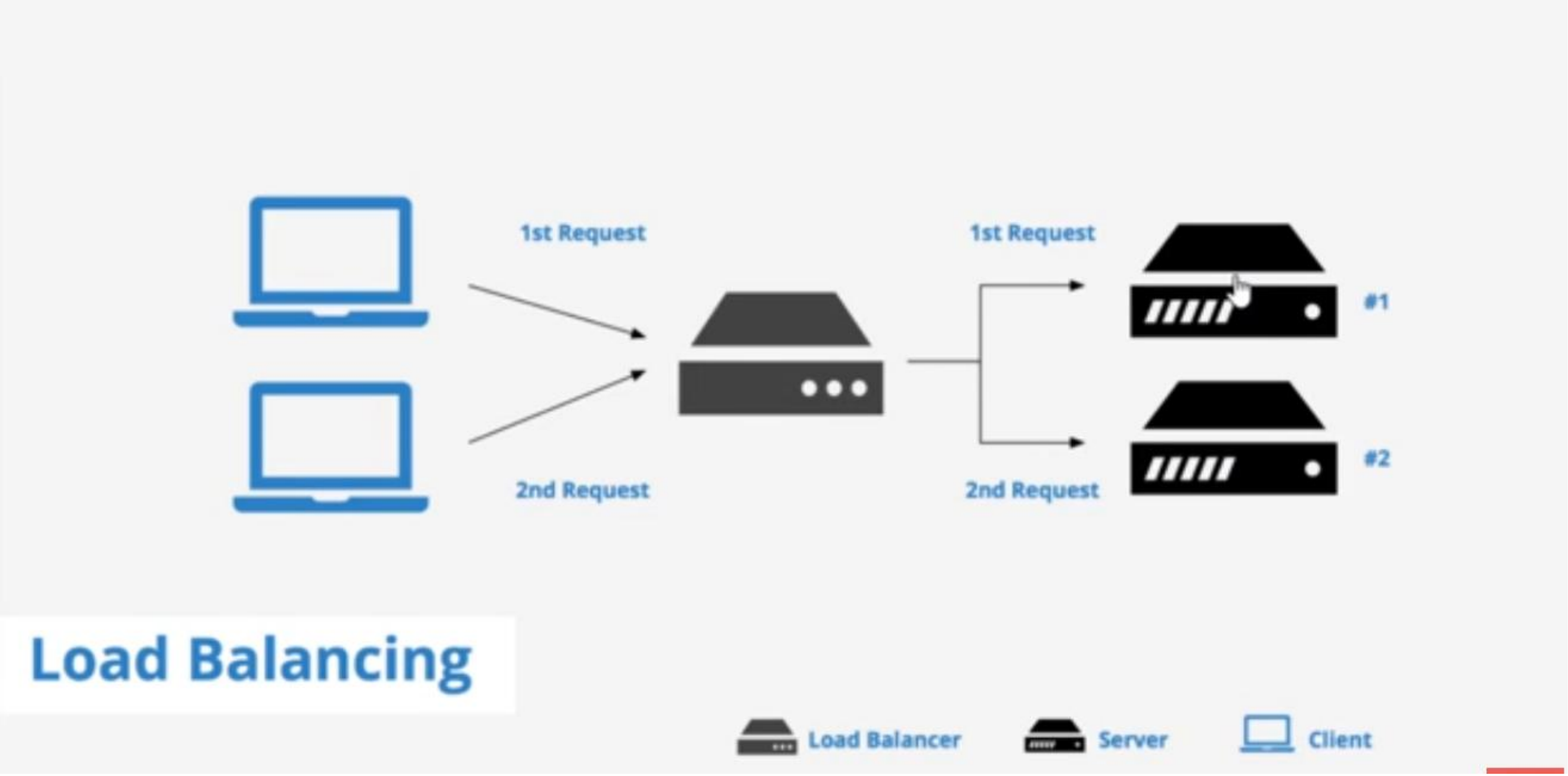
Load balancing is a computer networking method to distribute workload across multiple computers or a computer cluster, network links, central processing units, disk drives, or other resources.

Load Balancers: What is Load Balancer

- It helps to distribute load across multiple resources.
- It also keep track of status of all the resources while distributing requests. If a server is not available, it stops sending traffic.

Why load Balancing?

- Single Point of Entry
- Abstraction
- Scalability
- Reusability



Load Balancing

Load Balancers: Where it can be added

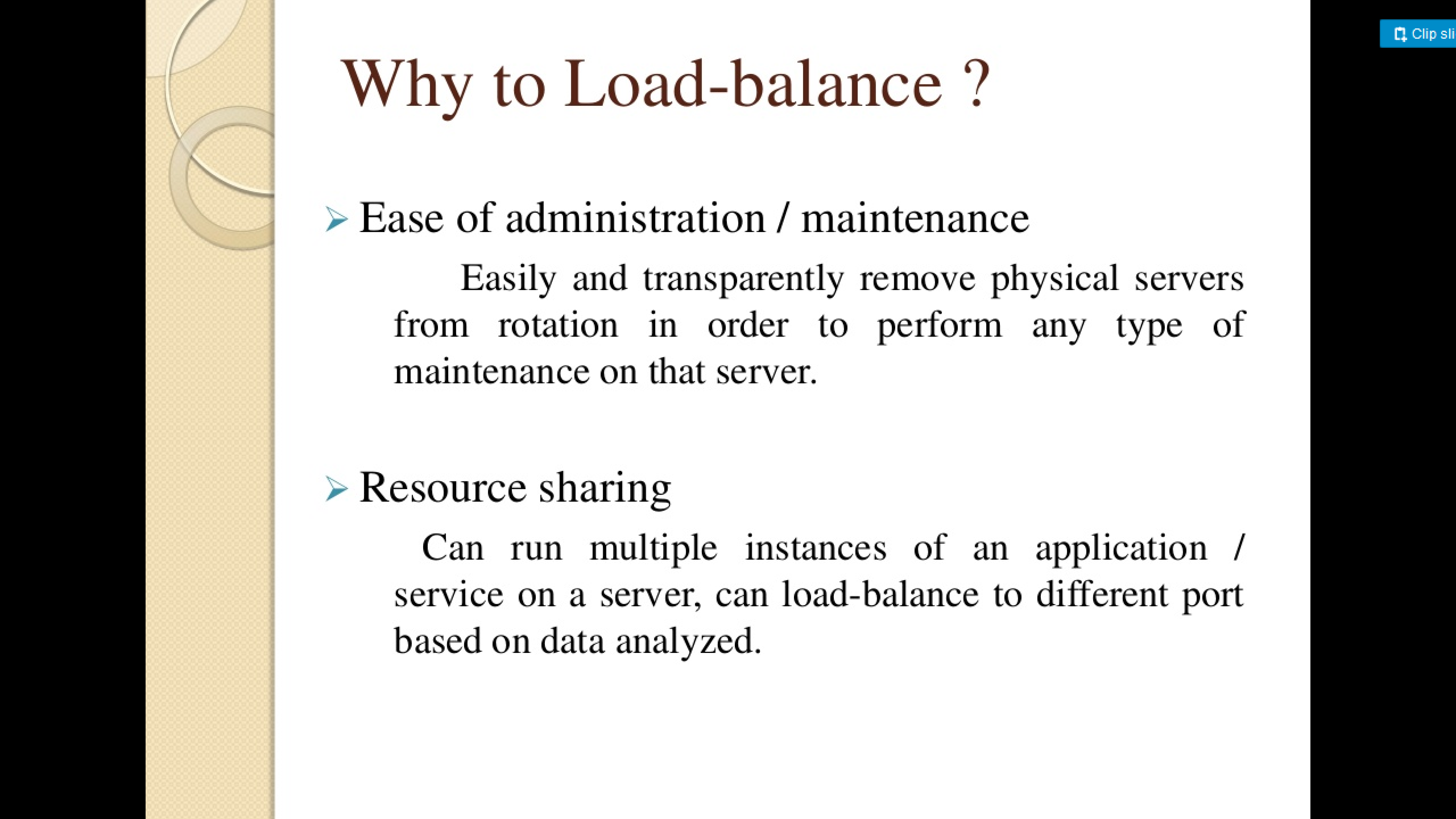
- User – Web Server
- Web server – Internal Server
- Internal Server - Database

Load Balancers: Types

- Hardware LB:
  - They are hardware which works as LB, but are very expensive.
  - Even big companies use them only as first point of contact & use other mechanism for load-balancing.
- Software Load balancers:
  - It's hybrid approach. HAProxy is popular open source software LB.
  - Every client request on this port (where HAProxy is running) will be received by proxy & then passed to the backend service in efficient way.

Goals of Load-Balancing

- Achieve optimal resource utilization
- Maximize throughput
- Minimize response time
- Avoid overload
- Avoid crashing



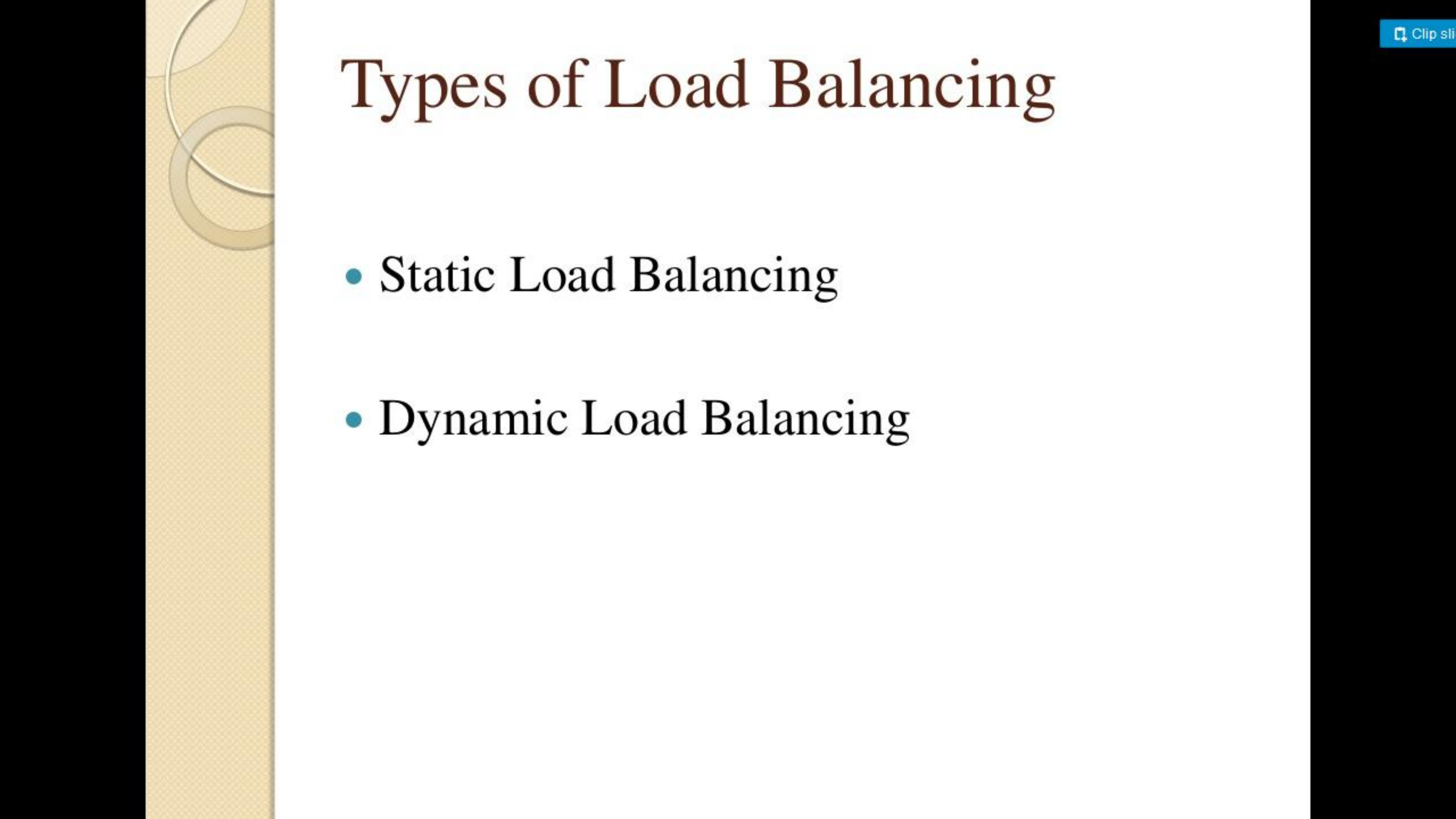
Why to Load-balance ?

- Ease of administration / maintenance

  Easily and transparently remove physical servers from rotation in order to perform any type of maintenance on that server.

- Resource sharing

  Can run multiple instances of an application / service on a server, can load-balance to different port based on data analyzed.



Types of Load Balancing

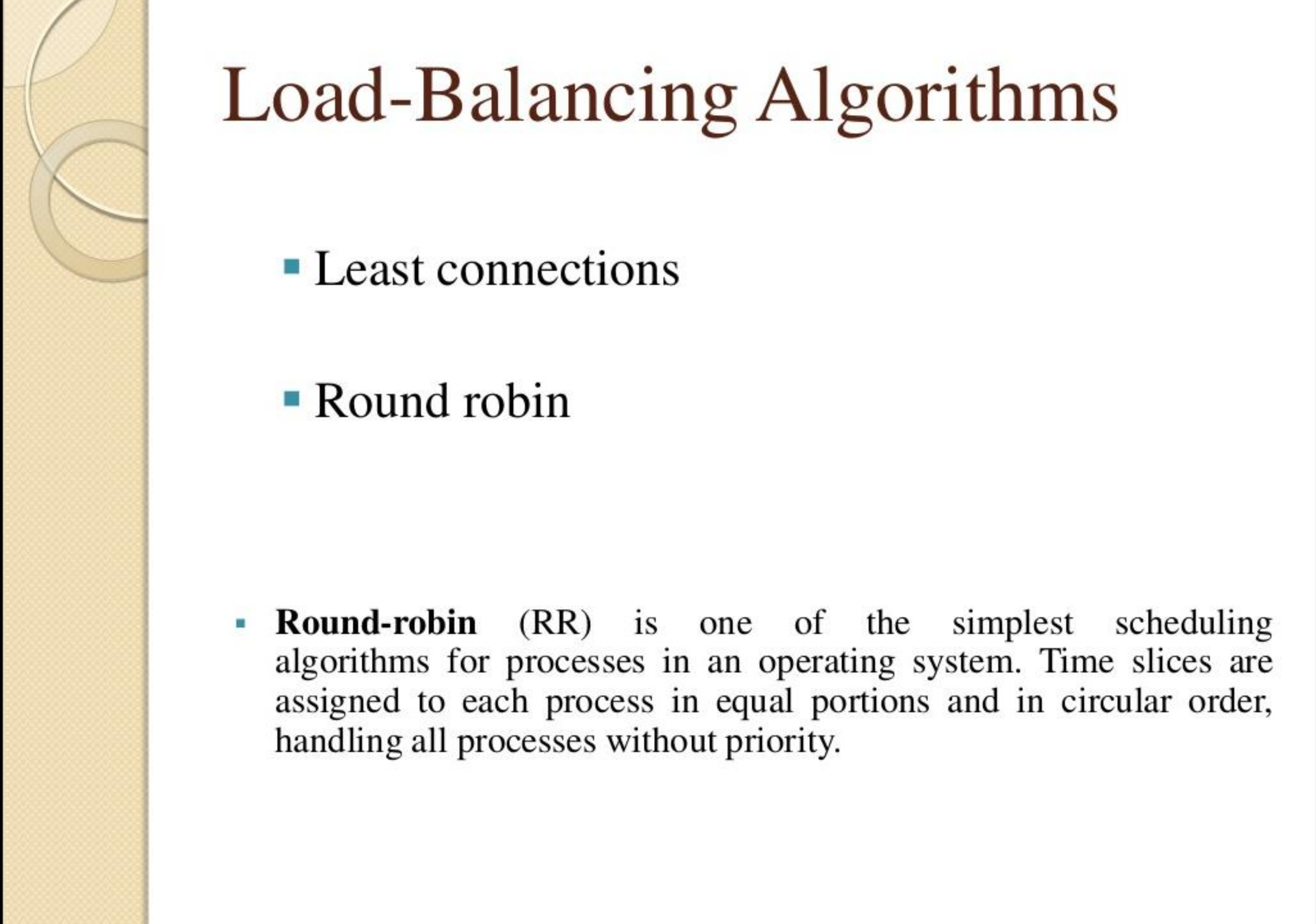
- Static Load Balancing
- Dynamic Load Balancing

Static Load Balancing

It is the type of Load Balancing which is often referred to as the mapping problem, the task is to map a static process graph onto a fixed hardware topology in order to minimise dilation and process load differences.

Dynamic Load Balancing

It is desirable in a distributed system to have the system load balanced evenly among the nodes so that the mean job response time is minimized.

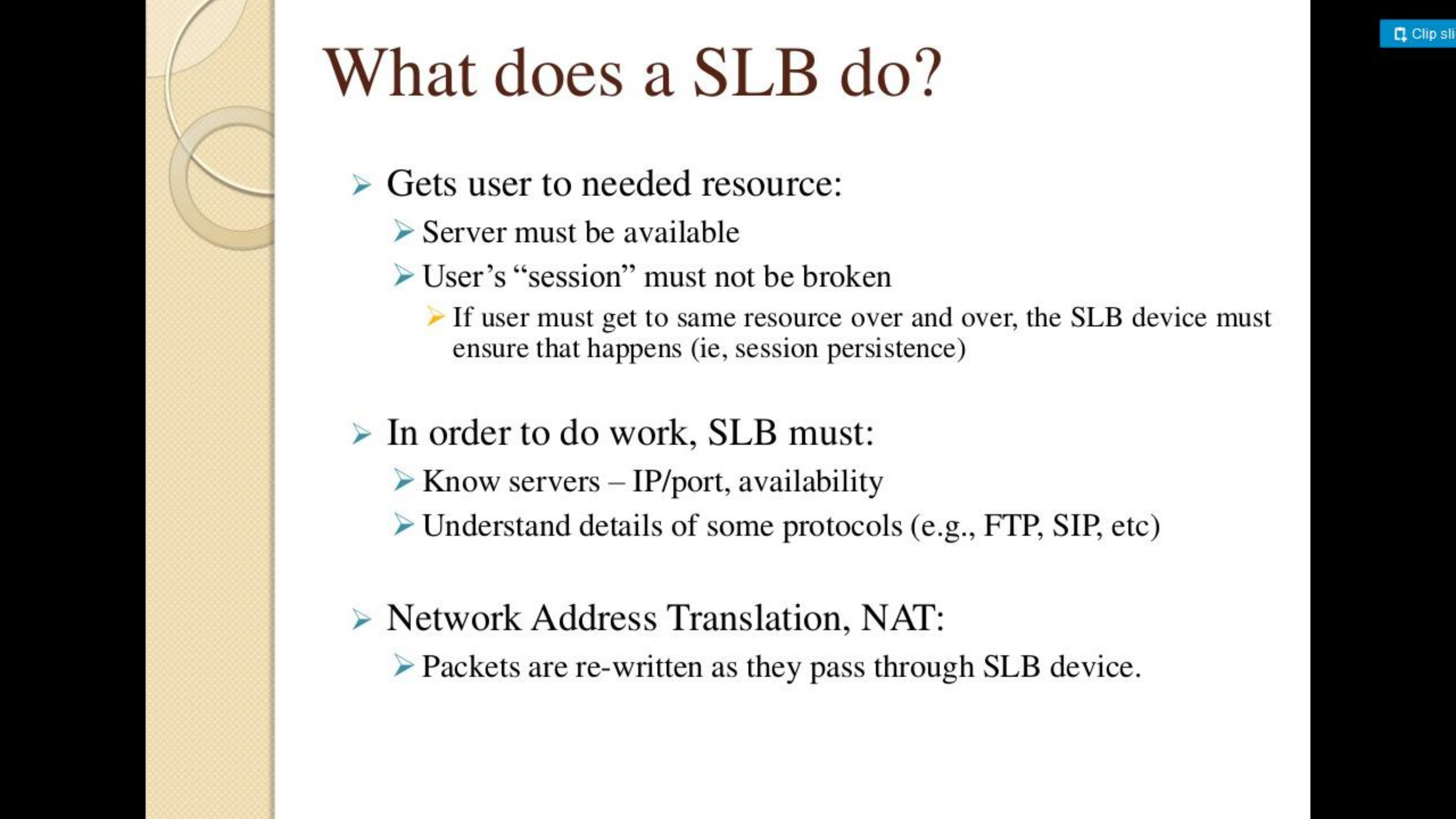


Load-Balancing Algorithms

- Least connections
- Round robin

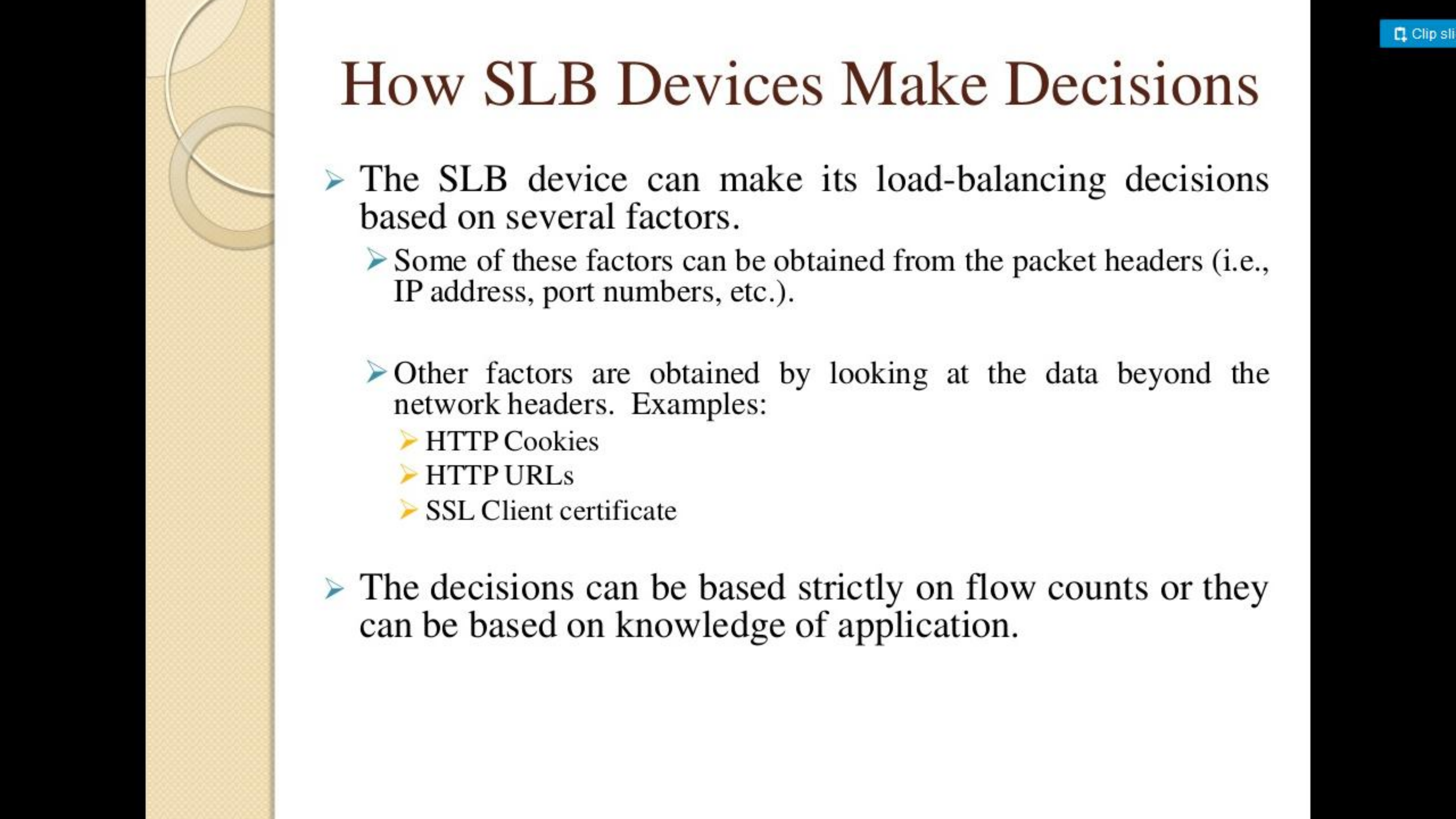
- **Round-robin** (RR) is one of the simplest scheduling algorithms for processes in an operating system. Time slices are assigned to each process in equal portions and in circular order, handling all processes without priority.

Server Load-balancing



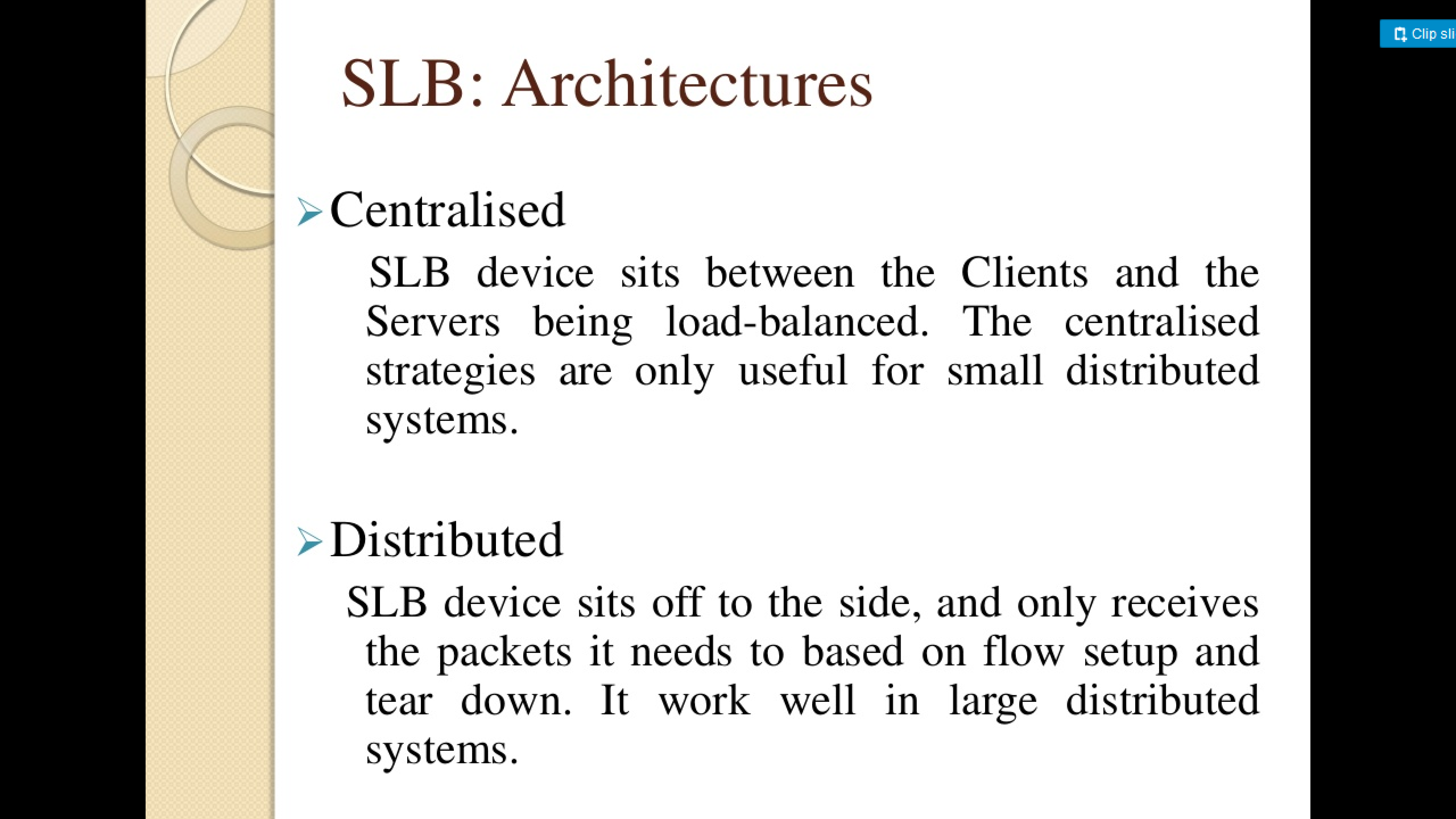
What does a SLB do?

- Gets user to needed resource:
  - Server must be available
  - User's "session" must not be broken
    - If user must get to same resource over and over, the SLB device must ensure that happens (ie, session persistence)
- In order to do work, SLB must:
  - Know servers – IP/port, availability
  - Understand details of some protocols (e.g., FTP, SIP, etc)
- Network Address Translation, NAT:
  - Packets are re-written as they pass through SLB device.



How SLB Devices Make Decisions

- The SLB device can make its load-balancing decisions based on several factors.
  - Some of these factors can be obtained from the packet headers (i.e., IP address, port numbers, etc.).
  - Other factors are obtained by looking at the data beyond the network headers. Examples:
    - HTTP Cookies
    - HTTP URLs
    - SSL Client certificate
- The decisions can be based strictly on flow counts or they can be based on knowledge of application.



SLB: Architectures

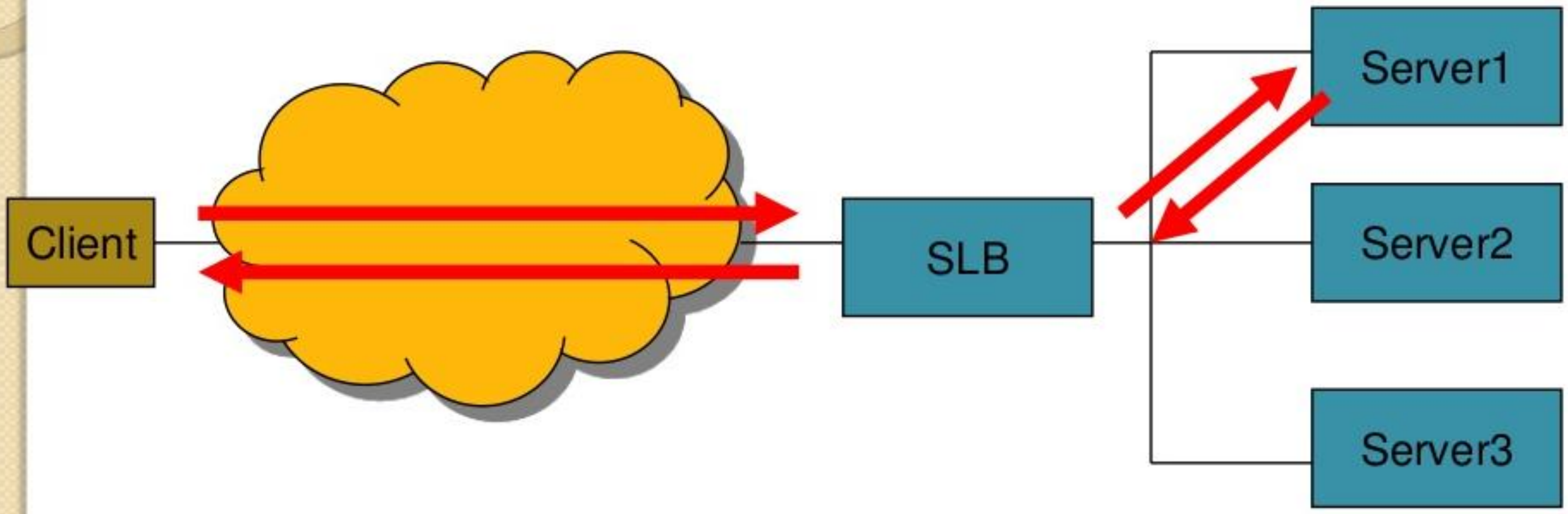
- Centralised

  SLB device sits between the Clients and the Servers being load-balanced. The centralised strategies are only useful for small distributed systems.

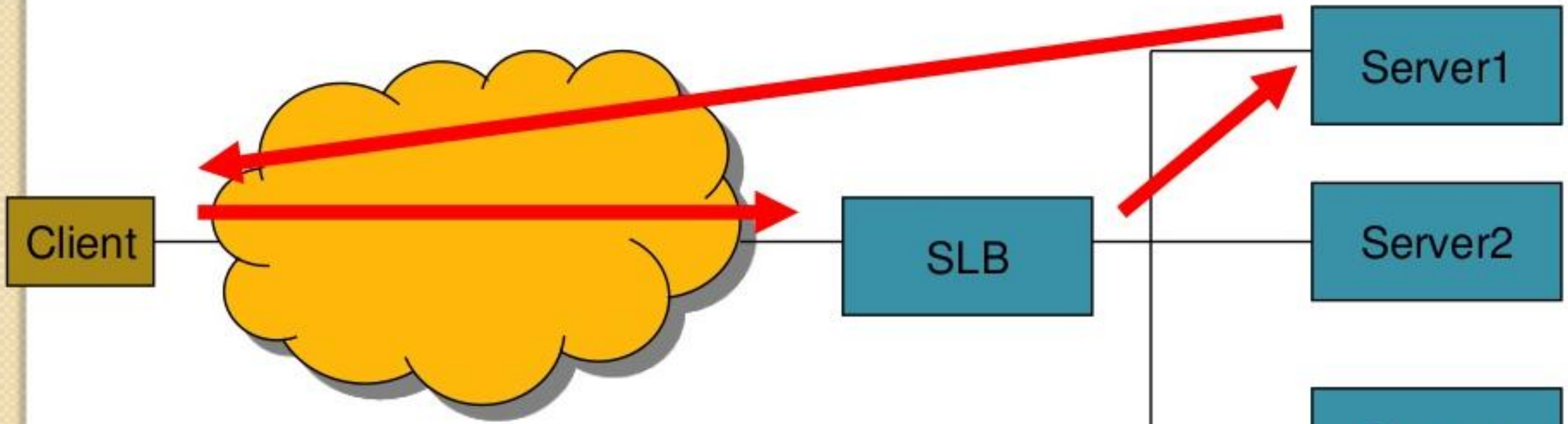
- Distributed

  SLB device sits off to the side, and only receives the packets it needs to based on flow setup and tear down. It work well in large distributed systems.

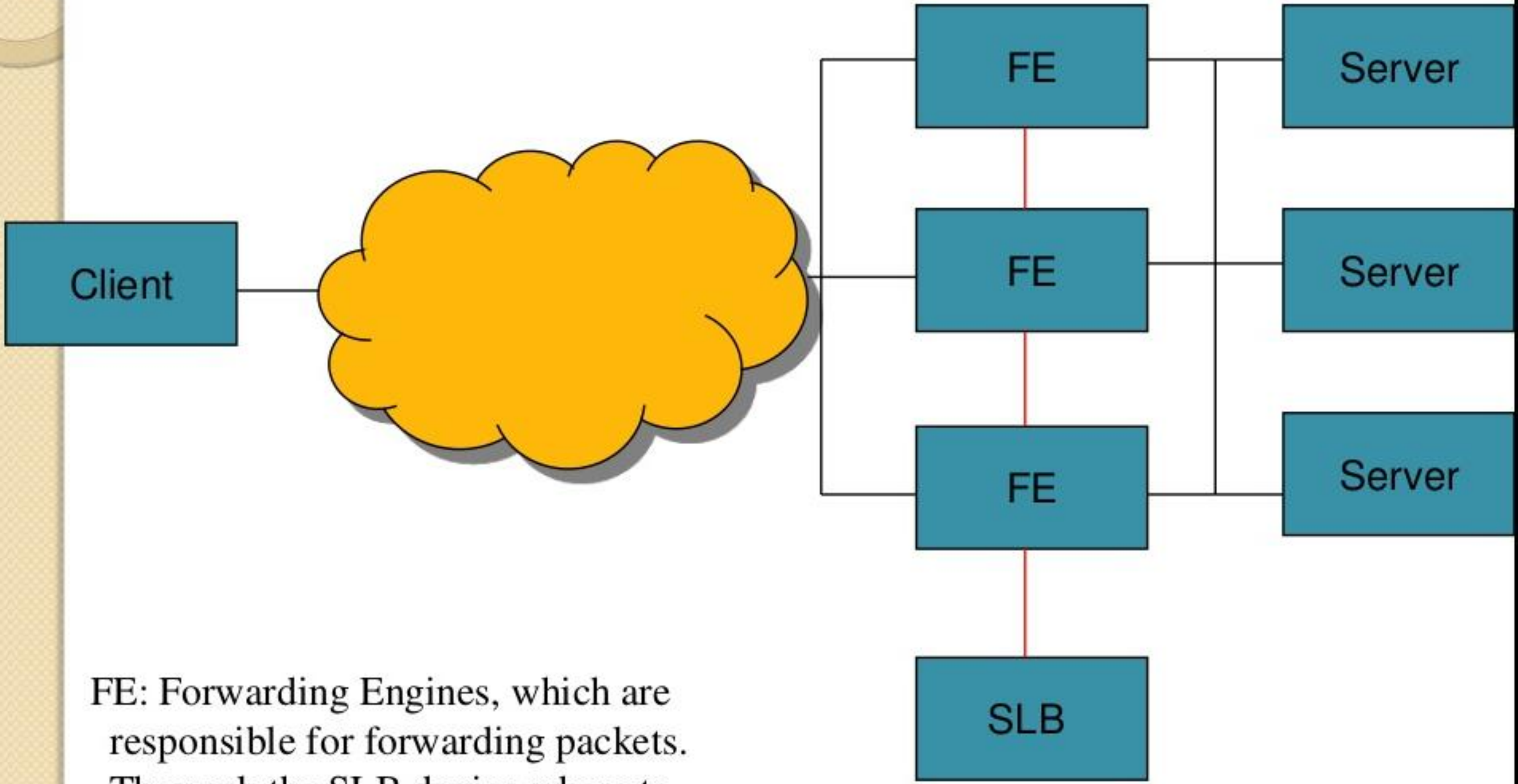
SLB: Centralised View with NAT



SLB: Centralised View without NAT

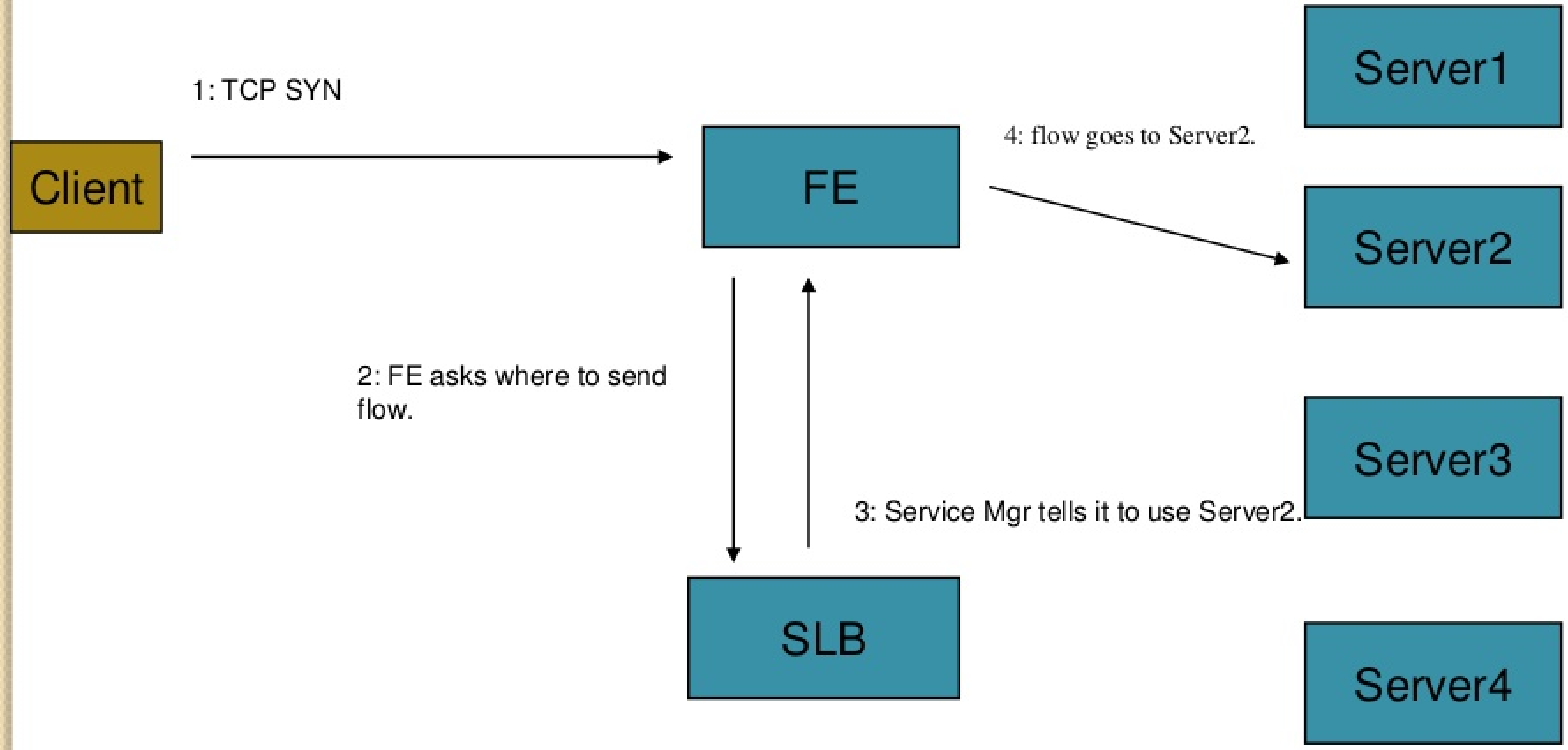


SLB: Distributed Architecture



FE: Forwarding Engines, which are responsible for forwarding packets. They ask the SLB device where to send the flow.

Distributed Architecture: Sample Flow



Client

1: TCP SYN

FE

4: flow goes to Server2.

Server1

Server2

Server3

Server4

2: FE asks where to send flow.

3: Service Mgr tells it to use Server2.

SLB

Subsequent packets flow directly from Client to Server2 thru the FE.
The FE must notify the SLB device when the flow ends.

Load Balancers: Algorithms

- Round Robin
- Round Robin with Weighted Server
- Least connections
- Least Response Time
- Source IP hash
- URL hash