Amirkabir University of Technology
(Tehran Polytechnic)

Cloud Computing

Introduction to Load Balancing

Seyyed Ahmad Javadi

sajavadi@aut.ac.ir

Spring 2021

Scheduling in Web Server Clusters

CS 260

LECTURE 3

From: IBM Technical Report

http://www.cs.ucr.edu/~bhuyan/CS260/index.html

Reference

The State of the Art in Locally Distributed Web-server Systems

Valeria Cardellini, Emiliano Casalicchio, Michele Colajanni and Philip S. Yu

Concepts

- Web server System

  Providing web services

  Trend:

  1. Increasing number of clients
  2. Growing complexity of web applications

- Scalable Web server systems

  The ability to support large numbers of accesses and resources while still providing adequate performance

Locally Distributed Web System

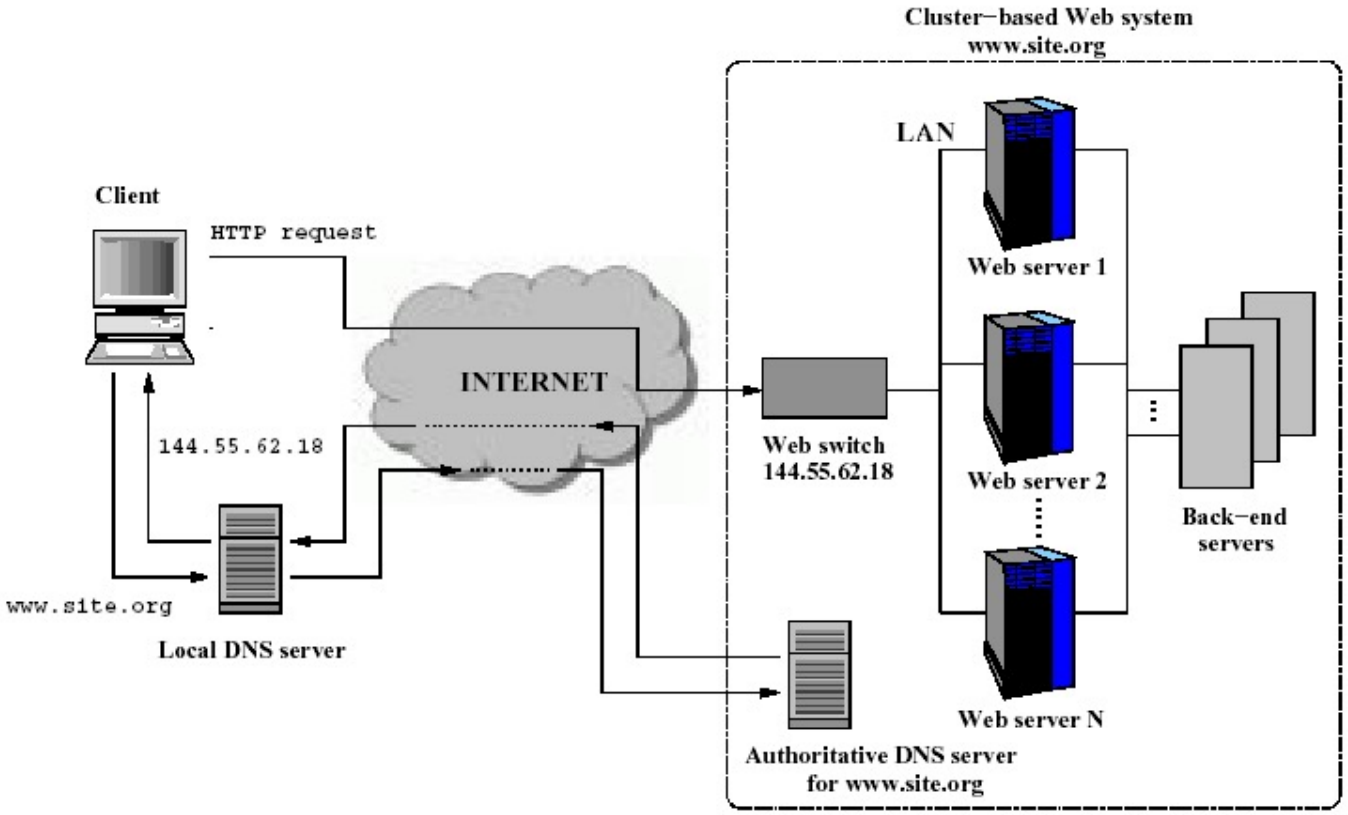
- Cluster Based Web System

  the server nodes mask their IP addresses to clients, using a Virtual IP address corresponding to one device (web switch) in front of the set of the servers – Web switch receives all packets and then sends them to server nodes.

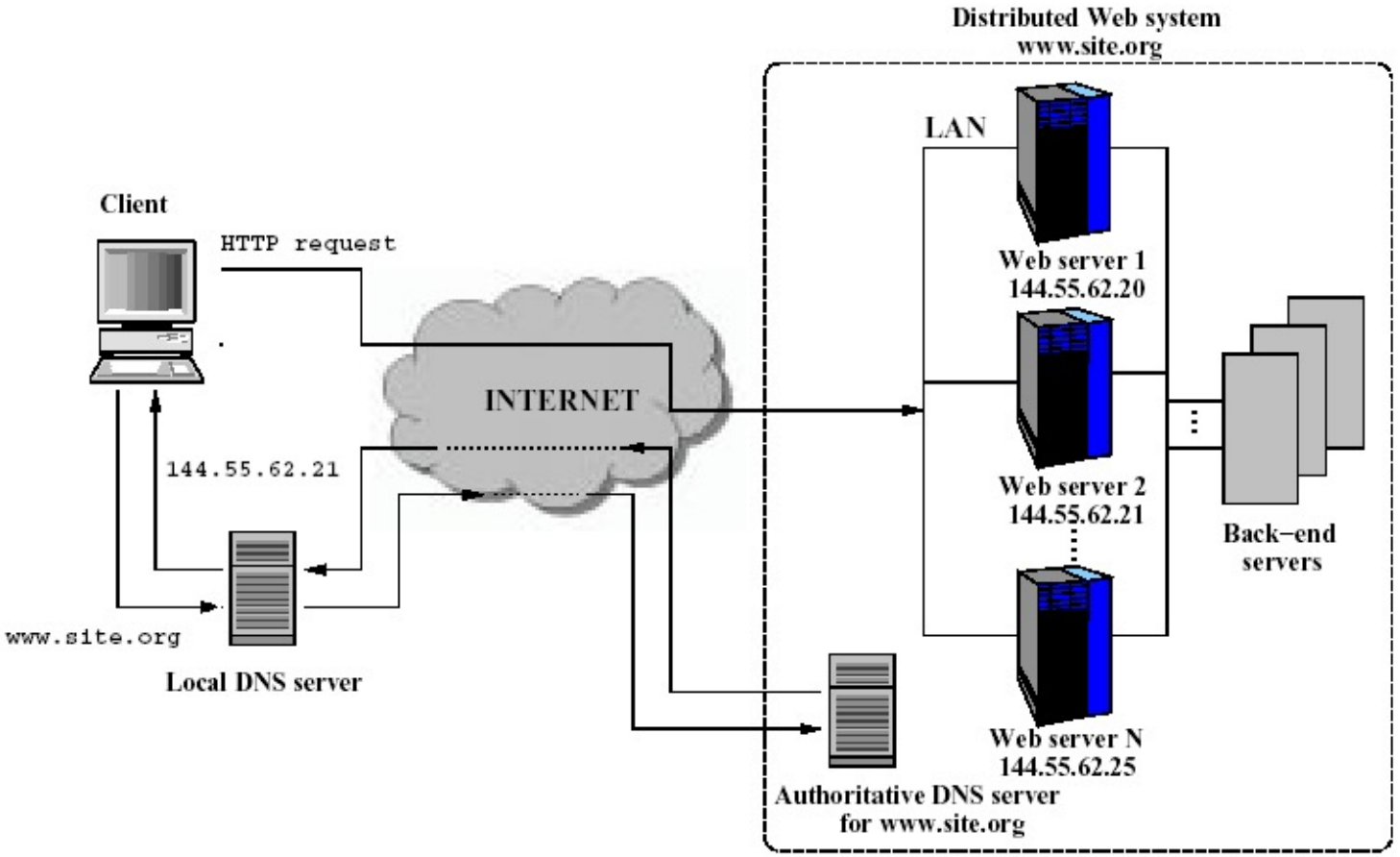
- Distributed Web System

  the IP addresses of the web server nodes are visible to clients. No web switch, just a layer 3 router may be employed to route the requests.

Cluster based Architecture



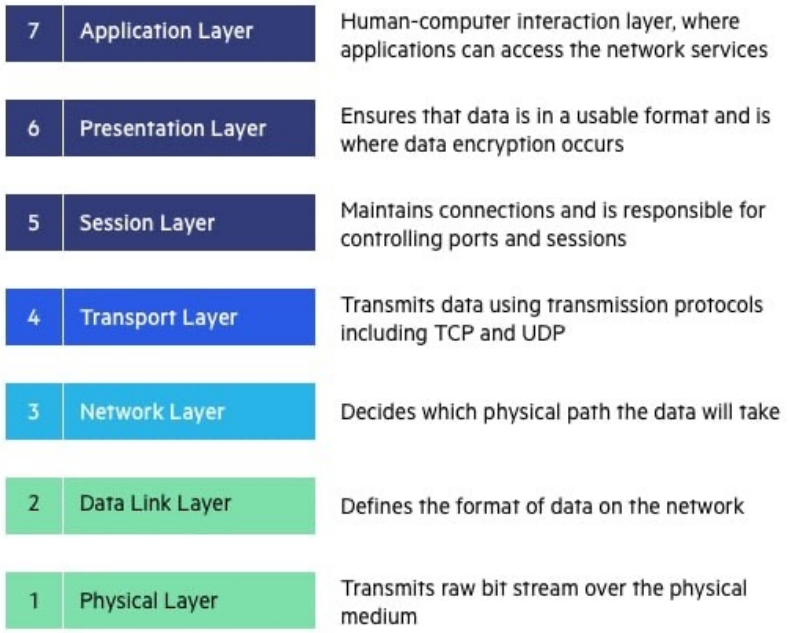
Distributed Architecture



Two Approaches

Depends on which OSI protocol layer at which the web switch routes inbound packets

- layer-4 switch – Determines the target server when TCP SYN packet is received. Also called *content-blind routing* because the server selection policy is not based on http contents at the application level
- layer-7 switch – The switch first establishes a complete TCP connection with the client, examines http request at the application level and then selects a server. Can support sophisticated dispatching policies, but large latency for moving to application level – Also called *Content-aware switches or Layer 5 switches* in TCP/IP protocol.



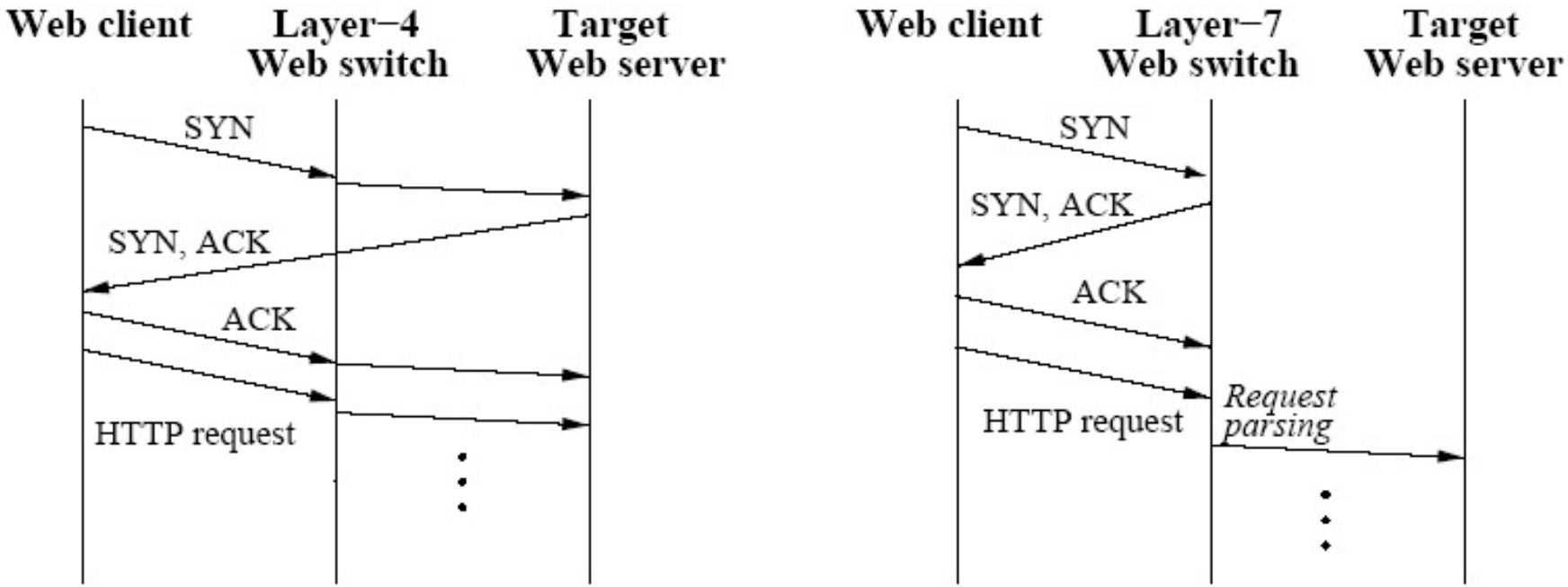
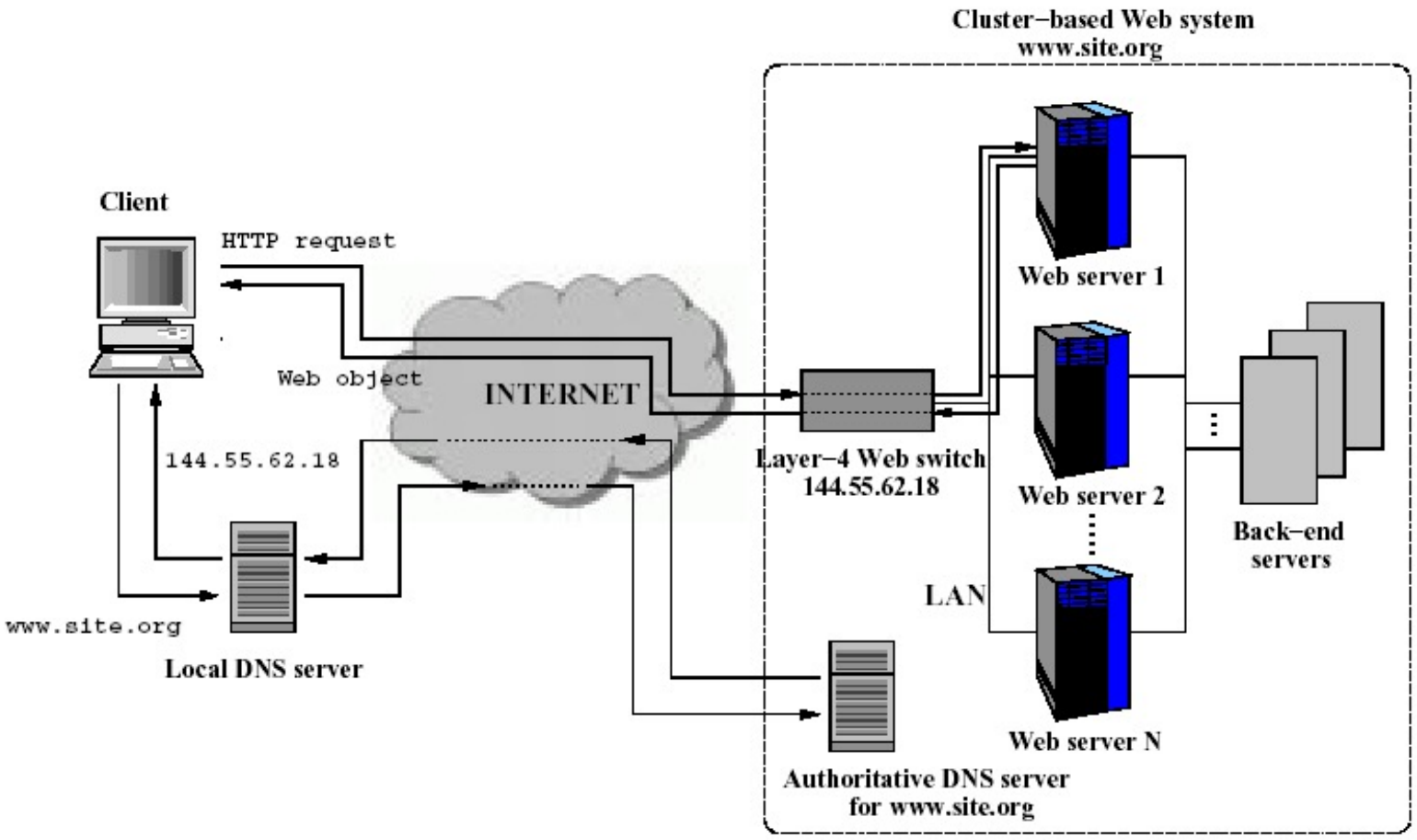
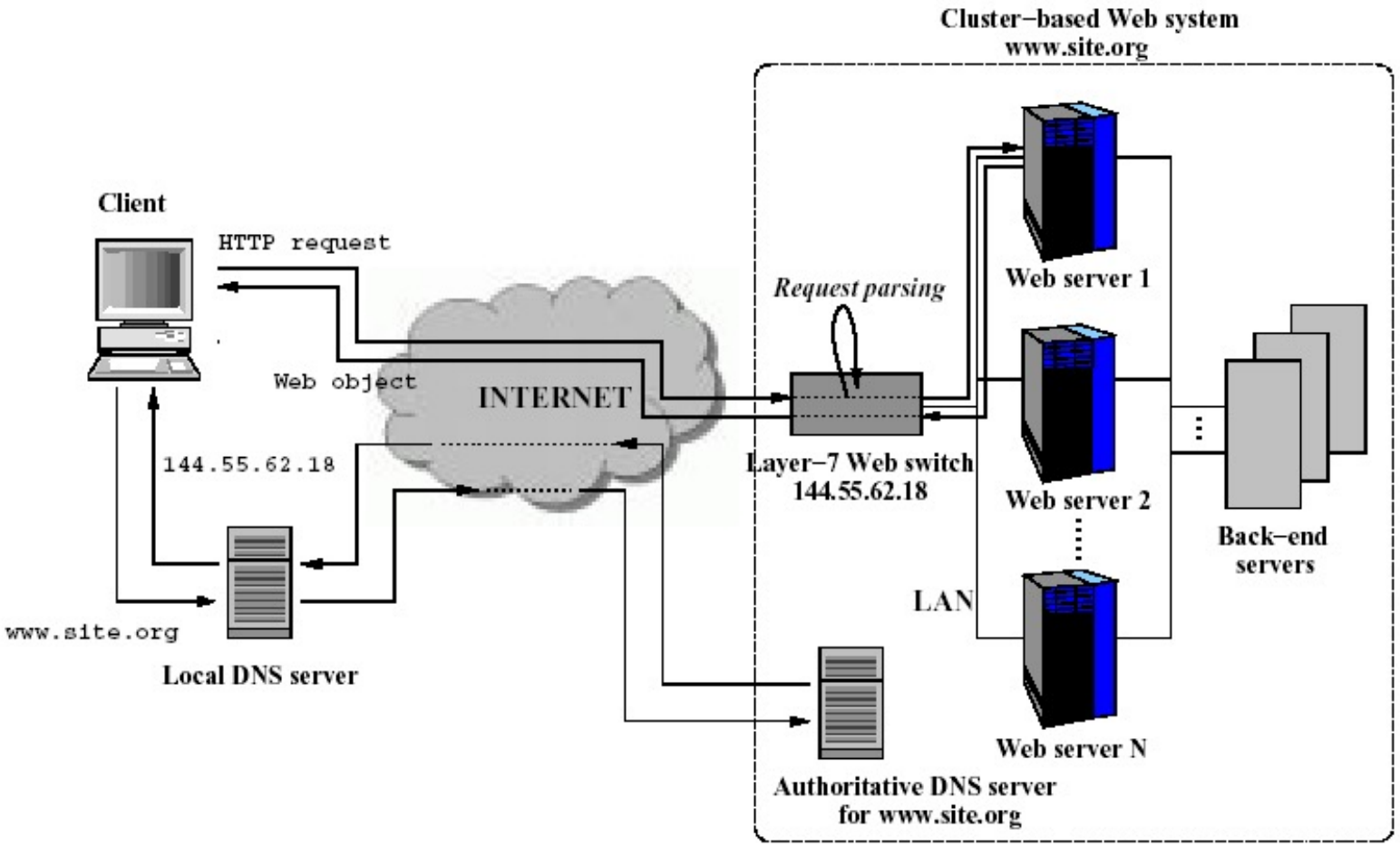


Figure 8: Operations of layer-4 routing (left) and layer-7 routing (right).

Layer-4 two-way architecture



Layer-7 two-way architecture



Layer-7 two-way mechanisms

- TCP gateway

  An application level proxy running on the web switch mediates the communication between the client and the server – makes separate TCP connections to client and server.

- TCP splicing

  Reduce the overhead in TCP gateway. For outbound packets, packet forwarding occurs at network level by rewriting the client IP address.

Layer-4 Products

Two-way	One-way		
Packet double-rewriting	*Packet single-rewriting*	*Packet tunneling*	*Packet forwarding*
Cisco's LocalDirector [33] Magicrouter [4] Linux Virtual Server [68] LSNAT [92] F5 Networks' BIG/ip [48] Foundry Networks' ServerIron [51] Cyber IQ's HyperFlow [39] HydraWEB's Hydra2500 [60] Coyote Point's Equalizer [37]	TCP Router [44]	Linux Virtual Server [68]	IBM Network Dispatcher [59, 61] Linux Virtual Server [68] ONE-IP [41] LSMAC [54] Intel's NetStructure Traffic Director [62] Nortel Networks' Alteon 780 [76] Foundry Networks' ServerIron [51] Radware's WSD Pro [85]

Layer 7 products

Two-way		One-way	
TCP gateway	*TCP splicing*	*TCP handoff*	*TCP connection hop*
IBM Network Dispatcher CBR [61]	[34]	ScalaServer [8, 79]	Resonate's Central Dispatch [86]
CAP [27]	Nortel Networks' Web OS SLB [76]		
HACC [101]	Foundry Networks' ServerIron [51]		
	Cisco's CSS [33]		
	F5 Networks' BIG/ip [48]		
	Radware's WSD Pro+ [85]		
	HydraWEB's Hydra2500 [60]		
	Zeus's Load Balancer [100]		
	[98]		

Dispatching Algorithms

Strategies to select the target server of the web clusters

- Static: Fastest solution to prevent web switch bottleneck, but **do not consider the current state of the servers.**

- Dynamic: Outperform static **algorithms by using intelligent decisions**, but collecting state information and analyzing them **cause expensive overheads.**

Requirements: (1) Low computational complexity (2) Full compatibility with web standards (3) state information must be readily available without much overhead.

Content blind approach

- Static Policies:

Random

distributes the incoming requests uniformly with equal probability of reaching any server

Round Robin (RR)

use a circular list and a pointer to the last selected server to make the decision

Static Weighted RR (For heterogeneous severs)

A variation of RR, where each server is assigned a weight Wi depending on its capacity

Content blind approach (Cont.)

- Dynamic

Client state aware

Static partitioning the server nodes and to assign group of clients identified through the clients information, such as source IP address

Server State Aware

Least Loaded, the server with the lowest load.

Issue: Which is the server load index?

Least Connection

fewest active connection first

Content blind approach (Cont.)

Server State Aware (Contd.)

- **Fastest Response**

  responding fastest

- **Weighted Round Robin**
  - Variation of static RR, associates each server with a dynamically evaluated weight that is proportional to the server load

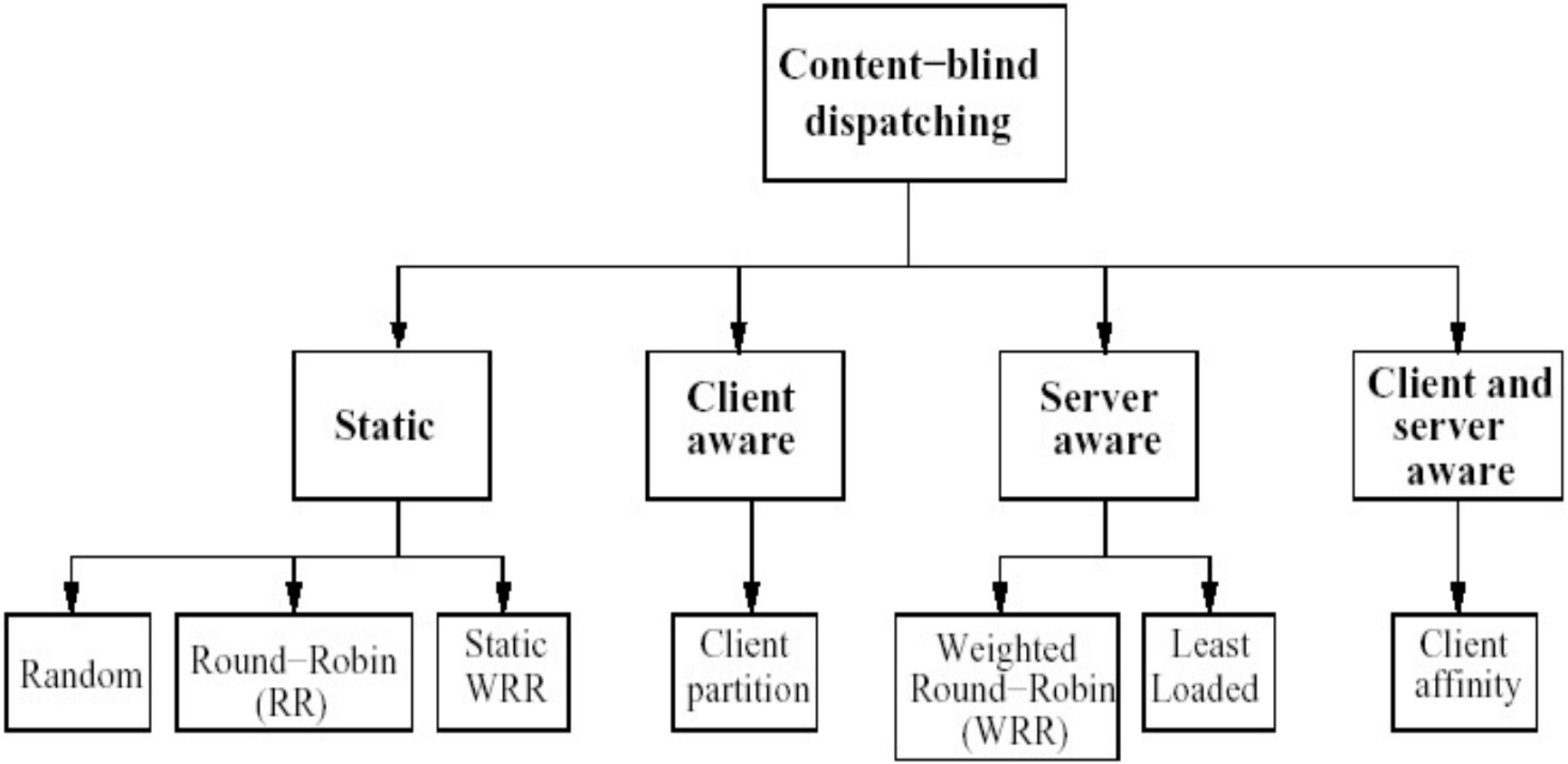
Client and server state aware

Client affinity

Instead of assigning each new connection to a server only on the basis of the server state regardless of any past assignment, consecutive connections from the same client can be assigned to the same server.

Considerations of content blind

- Static approach is the fastest, easy to implement, but **may make poor assignment decision**.
- Dynamic approach **has the potential to make better decision**, but it needs to **collect and analyze state information, may cause high overhead**.
- Overall, **simple server state aware algorithm is the best choice**, least loaded algorithm is commonly used in commercial products.


Content−blind dispatching
Static
Client aware
Server aware
Client and server aware
Random
Round−Robin (RR)
Static WRR
Client partition
Weighted Round−Robin (WRR)
Least Loaded
Client affinity

Content aware approach

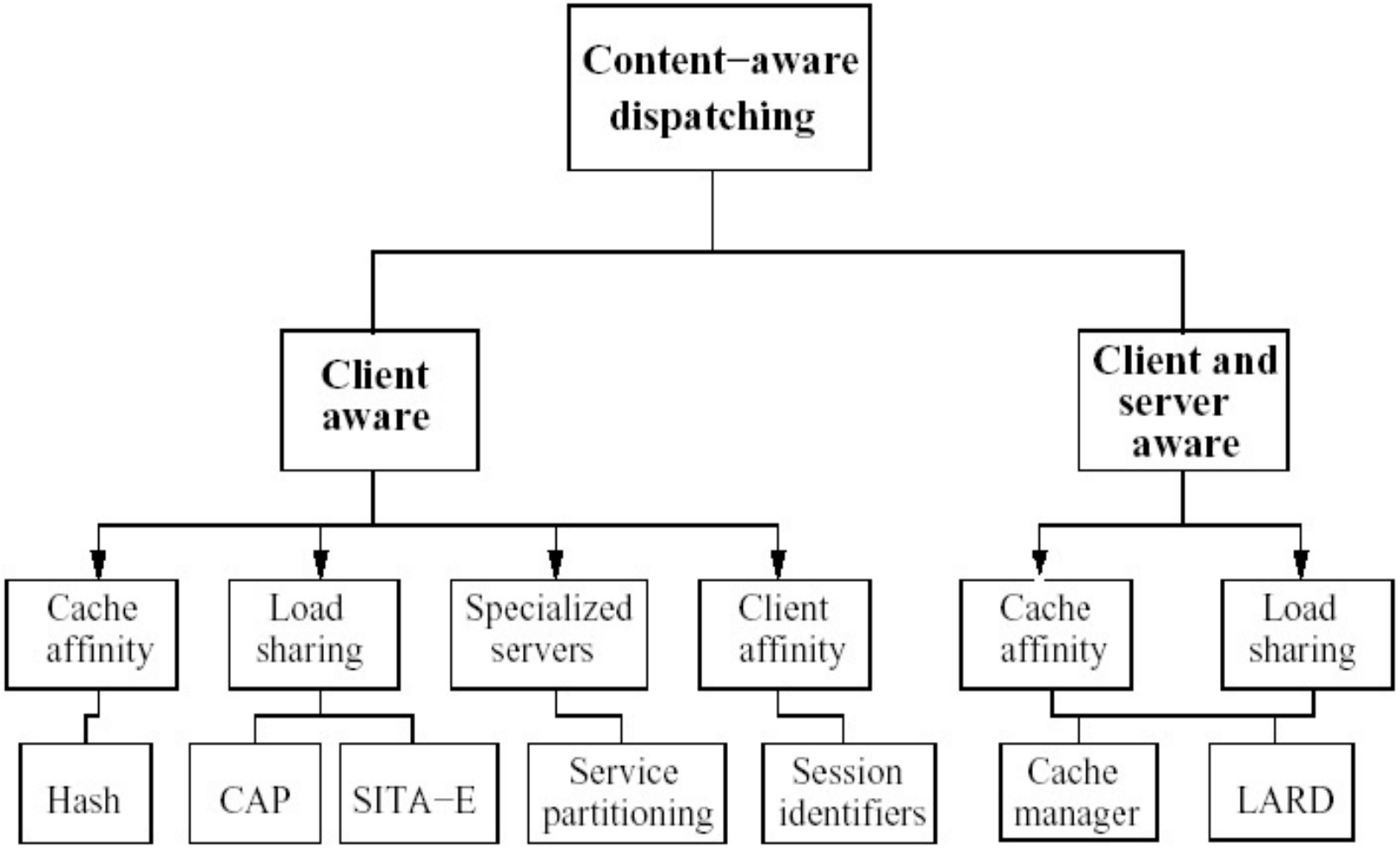
- **Sever state aware**
  - **Cache Affinity**
    - The file space is partitioned among the server nodes.
  - **Load Sharing**
    - **SITEA (Size Interval Task Assignment with Equal Load):** switch determines the size of the requested file and select the target server based on this information
    - **CAP (Client-Aware Policy):** web requests are classified based on their impact on system resources: such as I/O bound, CPU bound.

Content aware approach (Cont.)

- **Client state aware**
  - **Service Partitioning**
    - employ specialized servers for certain type of requests.
  - **Client Affinity**
    - using session identifier to assign all web transactions from the same client to the same server

Content aware approach (Cont.)

- **Client and server state aware**
  - **LARD (Locality aware request distribution**
    - direct all requests to the same web object to the same server node as long as its utilization is below a given threshold.
  - **Cache Manager**
    - A cache manager that is aware of the cache content of all web servers.


Content−aware dispatching
Client aware
Client and server aware
Cache affinity
Load sharing
Specialized servers
Client affinity
Cache affinity
Load sharing
Hash
CAP
SITA−E
Service partitioning
Session identifiers
Cache manager
LARD