School of Electronic Engineering - DCU

Switching and Systems Laboratory

Design and implementation of an MPLS based load balancing architecture for Web switching

Radu Dragos, Sanda Dragos and Martin Collier

RINCE | Research Institute for Networks and Communications Engineering

email: dragosr@eeng.dcu.ie
dragoss@eeng.dcu.ie
collierm@eeng.dcu.ie

1. Abstract

WWW - The preferred technology:

web server limitations ⇒

- New architectures - **Web clusters**
& New technologies - **Web Switching**

MultiProtocol Label Switching:

The preferred technology for:

- Traffic engineering
- Switching

The next generation of networks

Design & Implementation:

an MPLS web switching architecture for QoS enabled networks

1. Abstract

WWW - The preferred technology:
web server limitations ⇒
- New architectures - **Web clusters**
& New technologies - **Web Switching**

MultiProtocol Label Switching:
The preferred technology for:
- Traffic engineering
- Switching
The next generation of networks

Design & Implementation:
an MPLS web switching architecture for QoS enabled networks

1. Abstract

WWW - The preferred technology:

web server limitations ⇒

- New architectures - **Web clusters**

& New technologies - **Web Switching**

MultiProtocol Label Switching:

The preferred technology for:

- Traffic engineering
- Switching

The next generation of networks

Design & Implementation:

an MPLS web switching architecture for QoS enabled networks

1. Abstract

WWW - The preferred technology:

web server limitations ⇒

- New architectures - **Web clusters**

& New technologies - **Web Switching**

MultiProtocol Label Switching:

The preferred technology for:

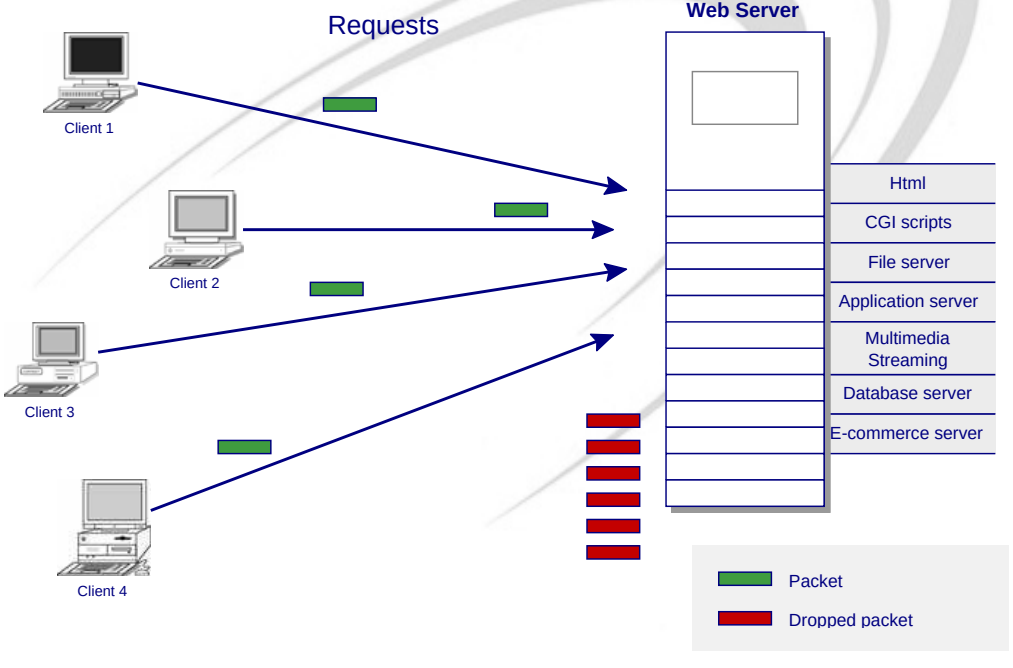
- Traffic engineering
- Switching

The next generation of networks

Design & Implementation:

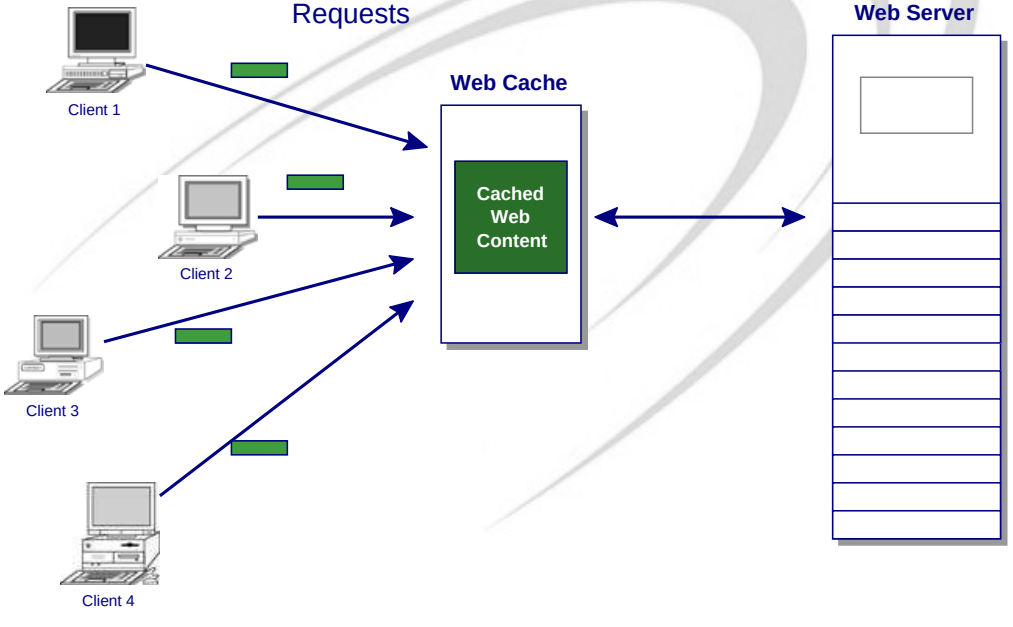
an MPLS web switching architecture for QoS enabled networks

2. Introduction



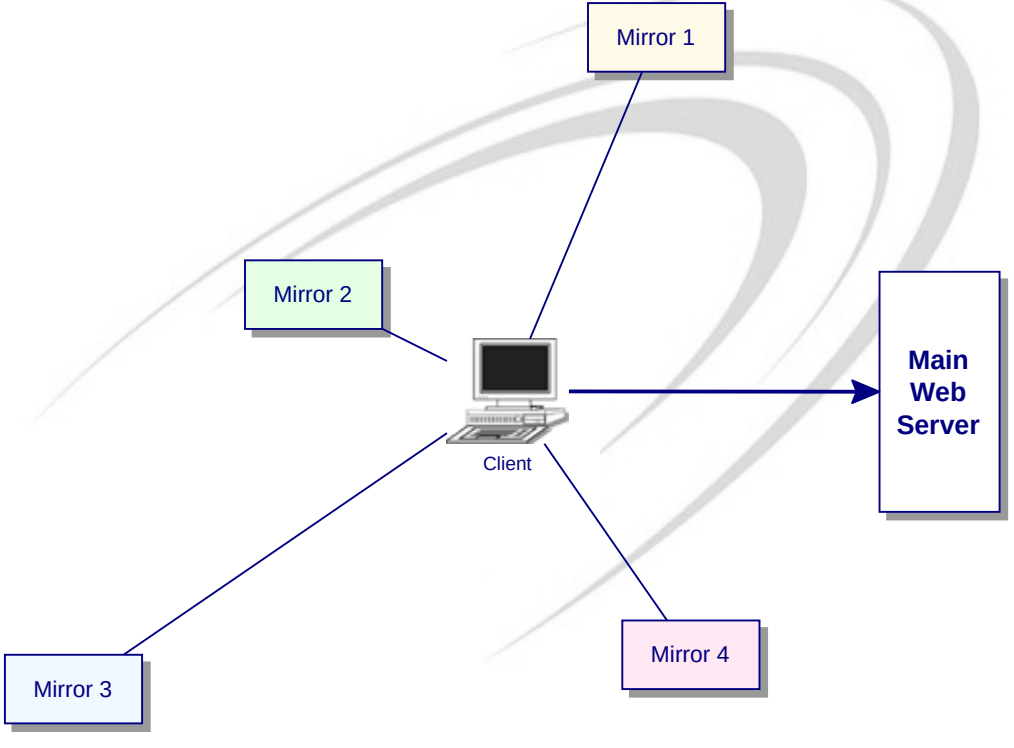
3. Solutions to the problem of over-congested Web servers

3.1. Web Content Caching



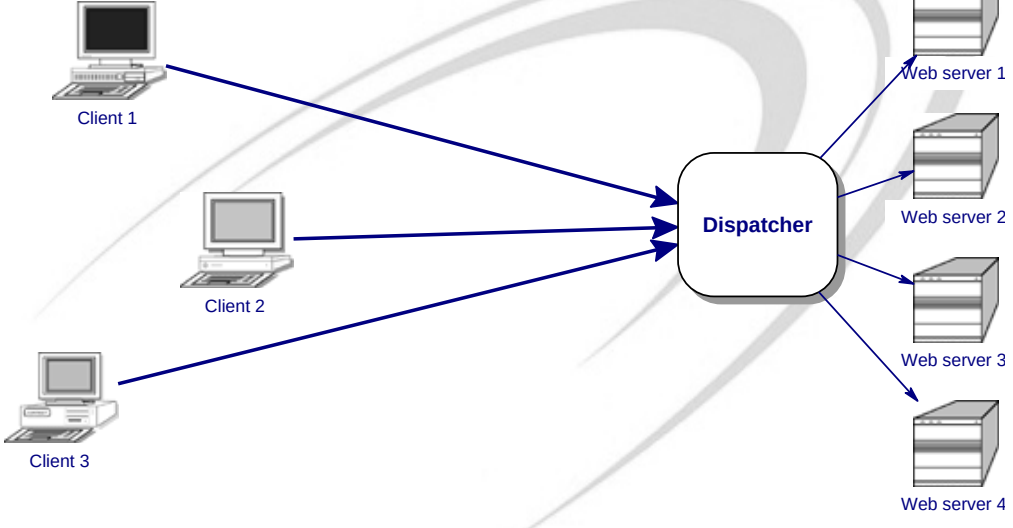
Limitations: *static* vs *dynamic* content

3.2. Web Content Mirroring



Drawbacks: *user decision*

3.3. Cluster of web servers



Categories: - *replicated content*
- *mirrored content*

4. Related problems

4.1. Overloading a web server

$\lambda \sim 10 connections/second$
$1/\mu \sim 30 seconds$

1 server: $C_1 \sim 300$

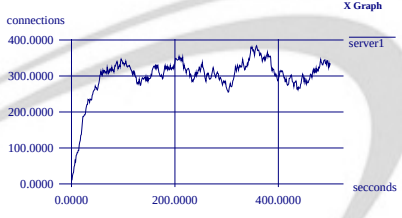
2 servers: $C_2 \sim 150$

3 servers: $C_3 \sim 100$

4. Related problems

4.1. Overloading a web server

$\lambda \sim 10 connections/second$
$1/\mu \sim 30 seconds$



1 server: $C_1 \sim 300$

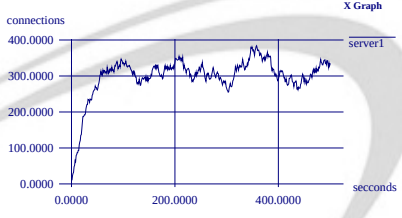
2 servers: $C_2 \sim 150$

3 servers: $C_3 \sim 100$

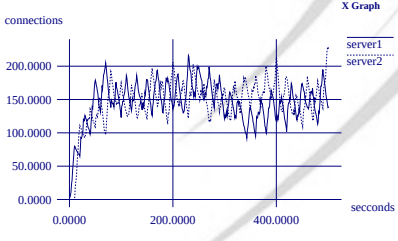
4. Related problems

4.1. Overloading a web server

$\lambda \sim 10 connections/second$
$1/\mu \sim 30 seconds$



1 server: $C_1 \sim 300$



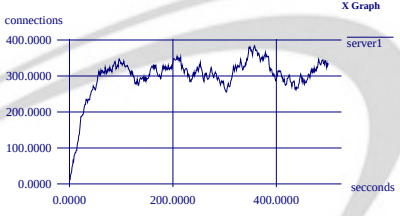
2 servers: $C_2 \sim 150$

3 servers: $C_3 \sim 100$

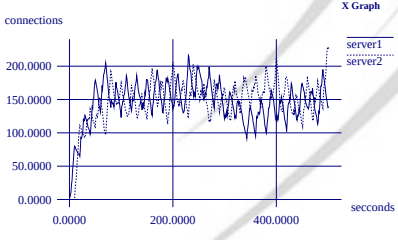
4. Related problems

4.1. Overloading a web server

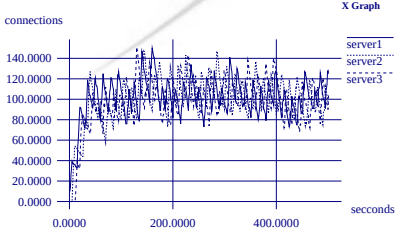
$\lambda \sim 10 connections/second$
$1/\mu \sim 30 seconds$



1 server: $C_1 \sim 300$

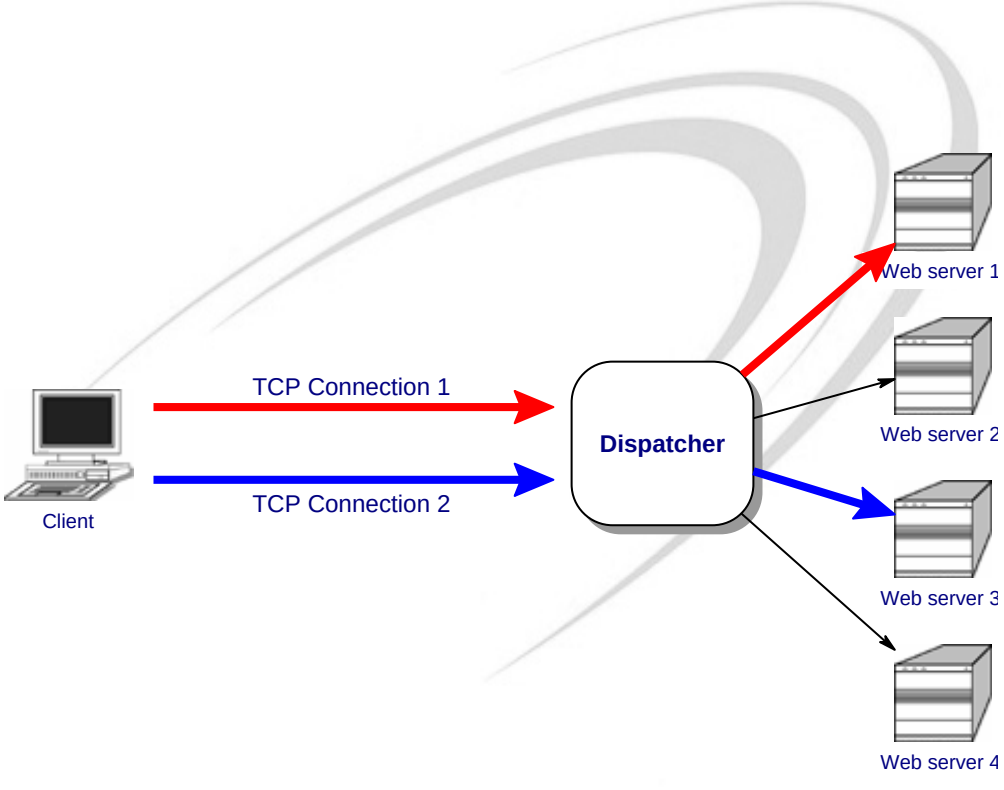


2 servers: $C_2 \sim 150$



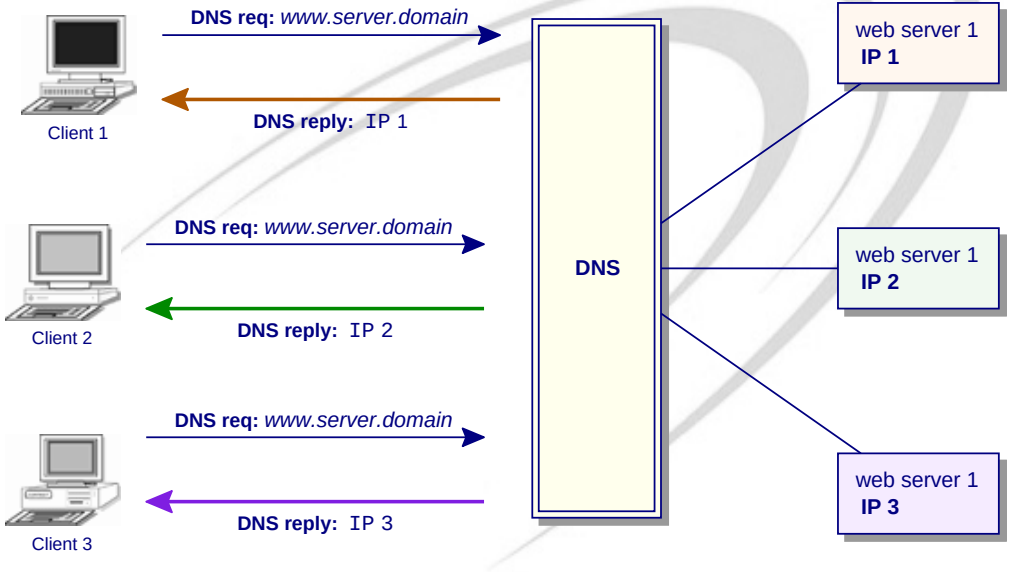
3 servers: $C_3 \sim 100$

4.2. The TCP continuity problem



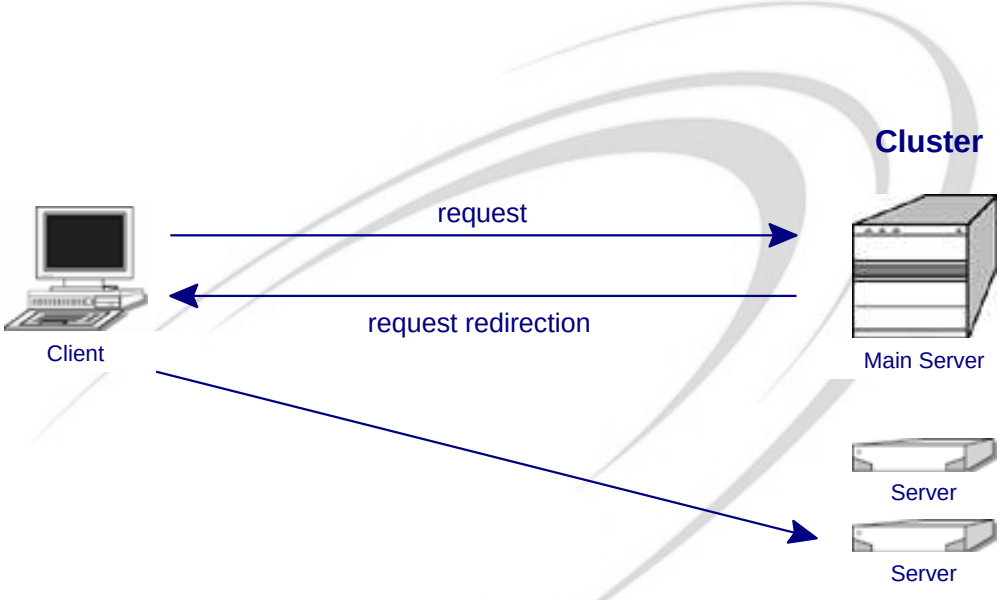
5. Approaches

5.1. Round-Robin DNS



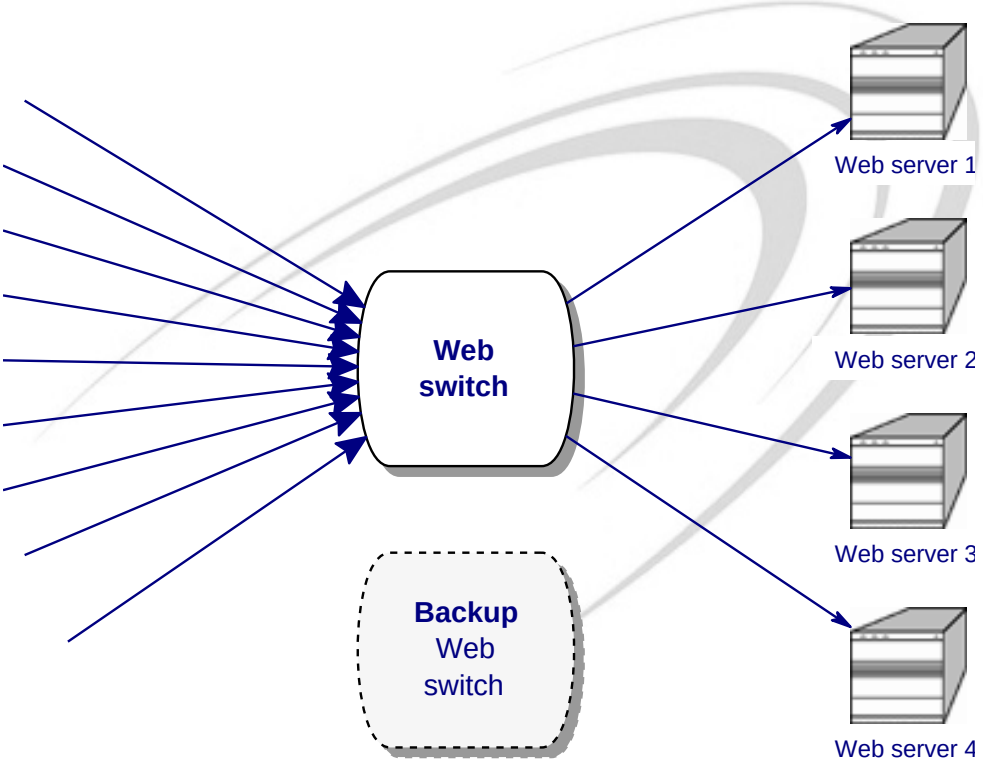
Limitations: - *DNS caching*
- *IP browsing*
- *load information*

5.2. HTTP redirect



Drawback: - *redirection delay*

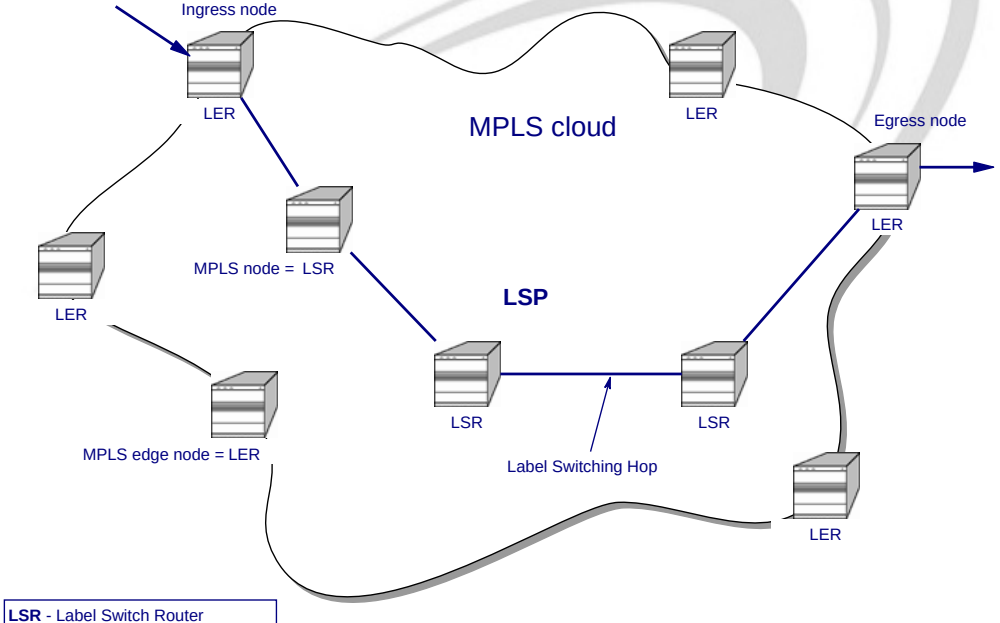
5.3. Load balancing switches



Drawbacks: - *single point of failure*
- *expensive solution*

6. MPLS Approach

6.1. Introduction to MPLS



6.2. Traffic Engineering objectives for MPLS environment

- traffic oriented

- resource oriented

6.2. Traffic Engineering objectives for MPLS environment

- traffic oriented

- resource oriented

6.2. Traffic Engineering objectives for MPLS environment

- traffic oriented

- resource oriented

Relation with load balancing:

6.2. Traffic Engineering objectives for MPLS environment

- traffic oriented

- resource oriented

Relation with load balancing:

▷ resource oriented objective
- server over- or under-utilization

6.2. Traffic Engineering objectives for MPLS environment

- traffic oriented

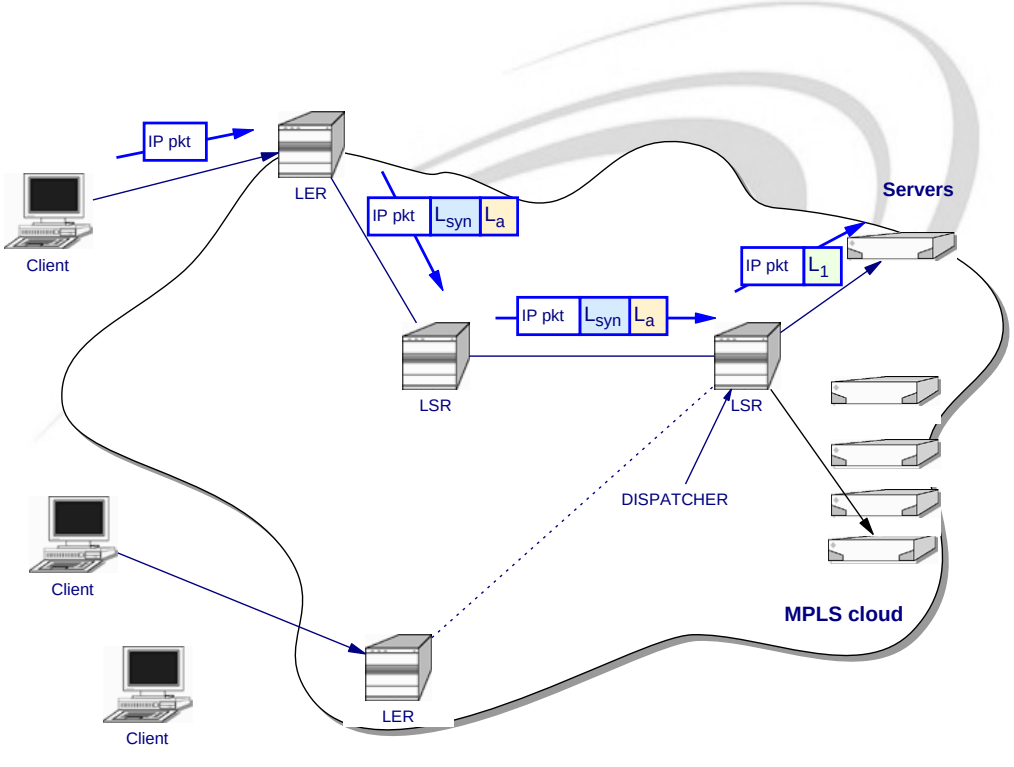
- resource oriented

Relation with load balancing:

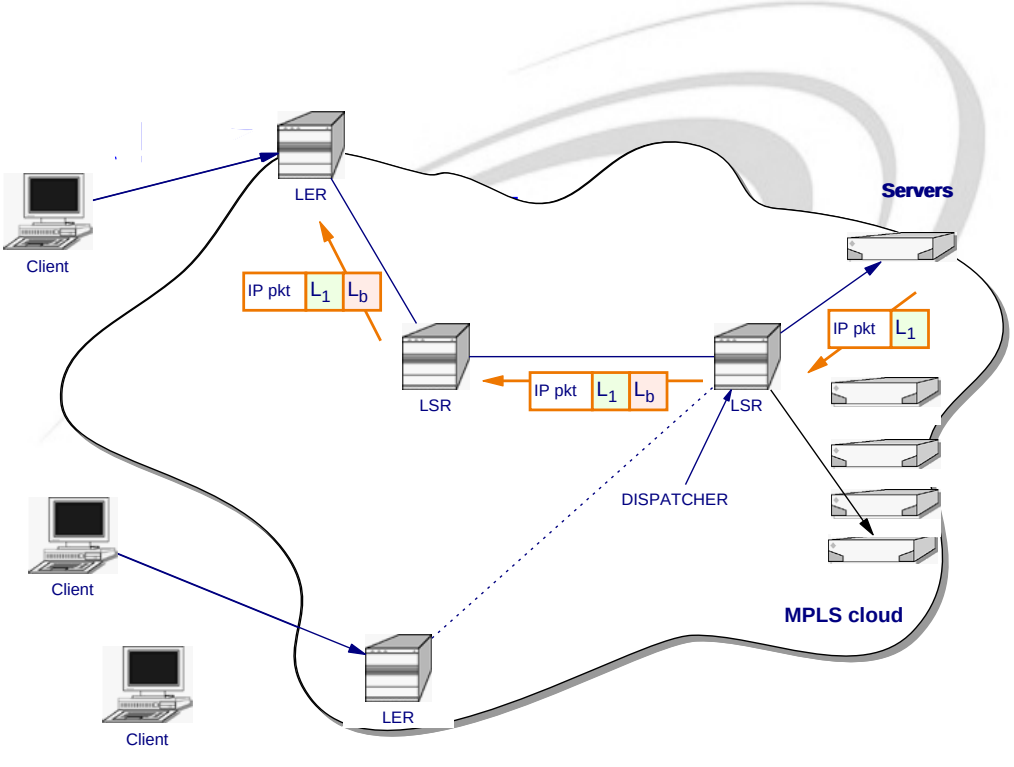
▷ resource oriented objective
- server over- or under-utilization

▷ traffic oriented objective
- supporting QoS operation

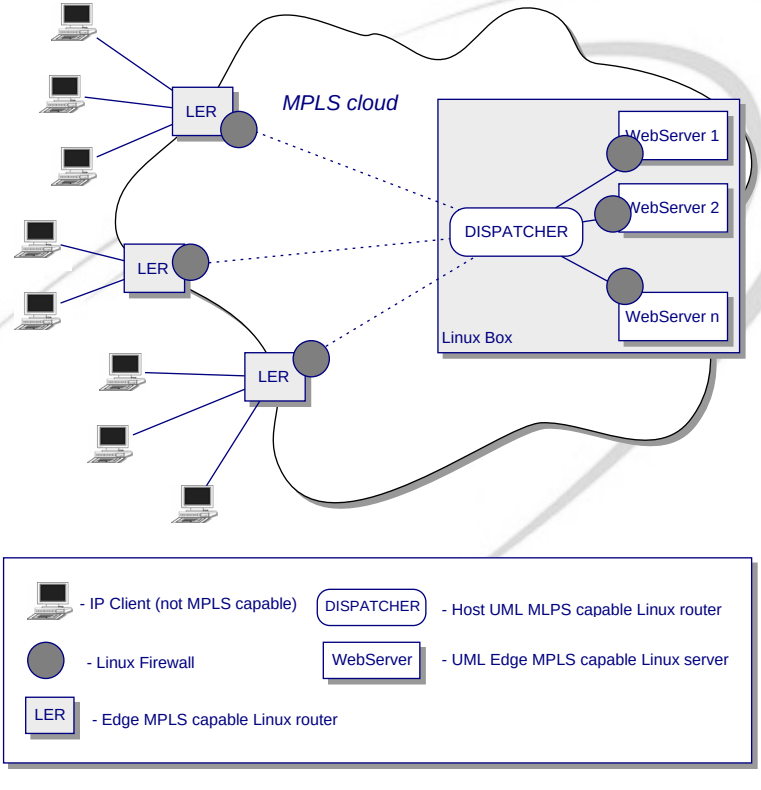
7. Framework



7. Framework



8. Implementation



9. Performance evaluation

9. Performance evaluation

Round-robin load balancing for large files:

	Server 1		Server 2		Server3	
	no.	%	no.	%	no.	%
20 con	7	35%	5	25%	8	40%
30 con	11	36.66%	12	40%	7	23.33%
50 con	16	32%	16	32%	18	36%

9. Performance evaluation

Round-robin load balancing for large files:

	Server 1		Server 2		Server3	
	no.	%	no.	%	no.	%
20 con	7	35%	5	25%	8	40%
30 con	11	36.66%	12	40%	7	23.33%
50 con	16	32%	16	32%	18	36%

Round-robin load balancing for small files:

	Server 1		Server 2		Server3	
	no.	%	no.	%	no.	%
300 con $\lambda = 3$	102	34%	99	33%	99	33%
900 con $\lambda = 9$	295	32.77%	303	33.66%	302	33.55%
300 con $\lambda = 12$	94	31.33%	96	32%	110	36.66%

10. Conclusions

- exploiting the QoS capabilities of underlaying network
- a novel technique for the next generation switching protocols (e.g. MPLS)
- a working, cost effective architecture in an open source environment
- MPLS based solution performed well even for highly loaded Web switches
  (12 connections/second $\sim$ 1 million hits/day)
- porting to hardware can increase the performance

10. Conclusions

- exploiting the QoS capabilities of underlaying network
- a novel technique for the next generation switching protocols (e.g. MPLS)
- a working, cost effective architecture in an open source environment
- MPLS based solution performed well even for highly loaded Web switches
  (12 connections/second $\sim$ 1 million hits/day)
- porting to hardware can increase the performance

10. Conclusions

- exploiting the QoS capabilities of underlaying network
- a novel technique for the next generation switching protocols (e.g. MPLS)
- a working, cost effective architecture in an open source environment
- MPLS based solution performed well even for highly loaded Web switches
  (12 connections/second $\sim$ 1 million hits/day)
- porting to hardware can increase the performance

10. Conclusions

- exploiting the QoS capabilities of underlaying network
- a novel technique for the next generation switching protocols (e.g. MPLS)
- a working, cost effective architecture in an open source environment
- MPLS based solution performed well even for highly loaded Web switches
  (12 connections/second $\sim$ 1 million hits/day)
- porting to hardware can increase the performance

10. Conclusions

- exploiting the QoS capabilities of underlaying network
- a novel technique for the next generation switching protocols (e.g. MPLS)
- a working, cost effective architecture in an open source environment
- MPLS based solution performed well even for highly loaded Web switches
  (12 connections/second $\sim$ 1 million hits/day)
- porting to hardware can increase the performance

10. Conclusions

- exploiting the QoS capabilities of underlaying network
- a novel technique for the next generation switching protocols (e.g. MPLS)
- a working, cost effective architecture in an open source environment
- MPLS based solution performed well even for highly loaded Web switches
  (12 connections/second $\sim$ 1 million hits/day)
- porting to hardware can increase the performance