HYDRANET-FT:
Network Support for Dependable Services

Gurudatt Shenoy   Suresh K. Satapati
Riccardo Bettati

Department of Computer Science
Texas A&M University

Failure Scenarios

"Failures ain't what they used to be."

		disaster type		
		system	site	service
"crash"	host	▓	■	
	network	▓	▓	■
overload	server	□	▓	■
	network	□	▓	░
DOS attack	local	□	▓	░
	distributed	□	▓	▓

Good Solutions to Support Highly Tolerant Services...

Programmatic Requirements

… provide service replication across geographically distributed server hosts.

… can be deployed and recalled on demand.

… preserve server control over data.

… support any service.

… can be incrementally deployed.

… remain fully transparent to clients.

both <u>mechanisms</u> and <u>recovery</u>!

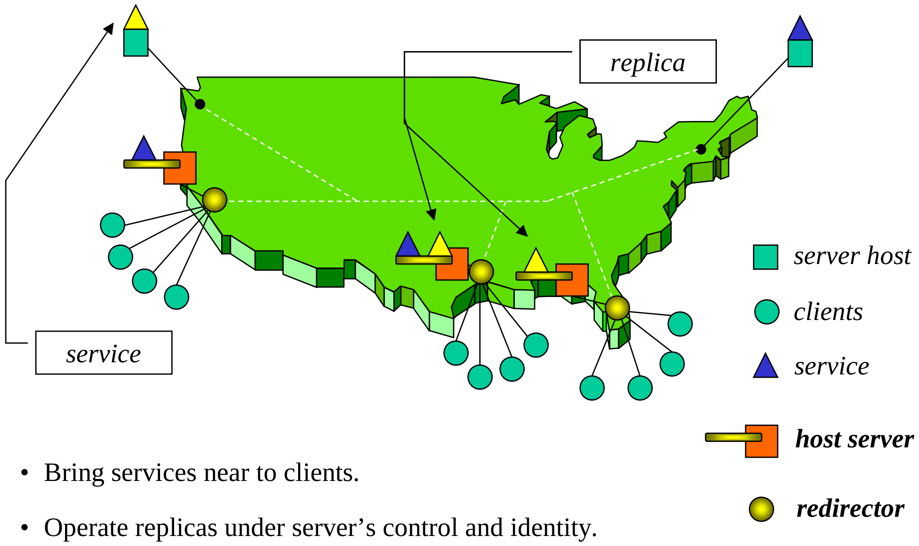
Tolerance Requirements

… support atomic multicast.

… ensure message ordering.

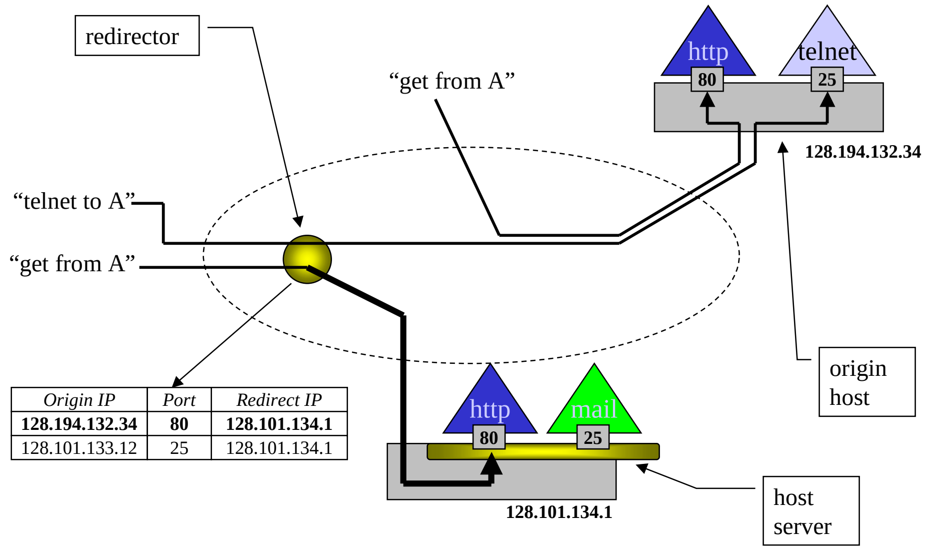
… provide low-latency failure detection mechanisms.

The HYDRANET Approach to Redirection/Replication



- Bring services near to clients.
- Operate replicas under server’s control and identity.

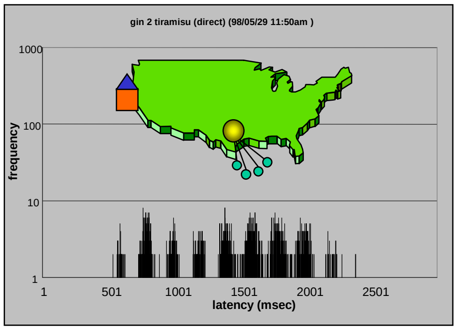
Mechanisms for Redirection



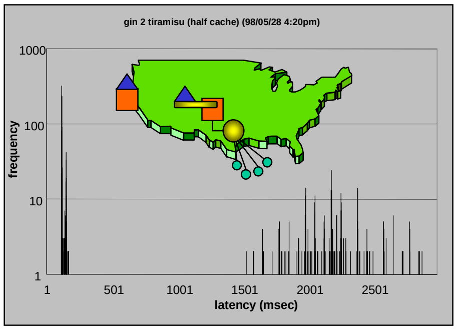
Origin IP	*Port*	*Redirect IP*
128.194.132.34	**80**	**128.101.134.1**
128.101.133.12	25	128.101.134.1

Performance Gains through HYDRAWEB Replication

Access latencies for 1kB HTTP objects.



No Replication

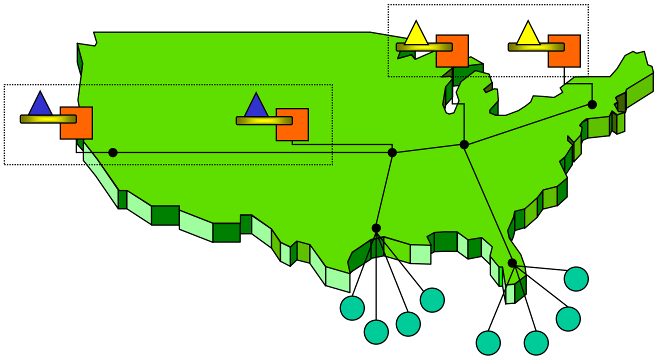


HYDRAWEB Replication

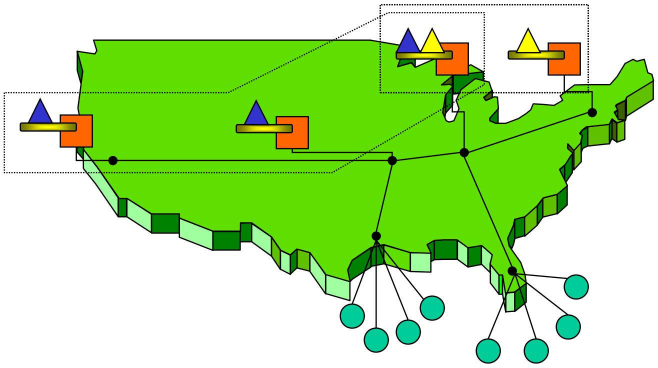
Origin host: `berkeley.edu`
Client host: `tamu.edu`

Origin host: `berkeley.edu`
Client host: `tamu.edu`
Host server: `tamu.edu`

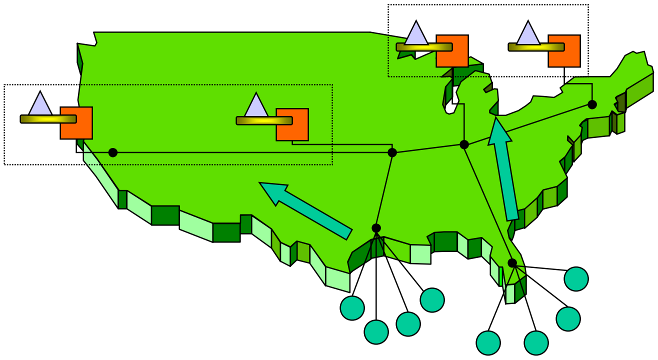
The HYDRANET-FT Approach to Support of Highly-Tolerant Services



The HYDRANET-FT Approach to Support of Highly-Tolerant Services



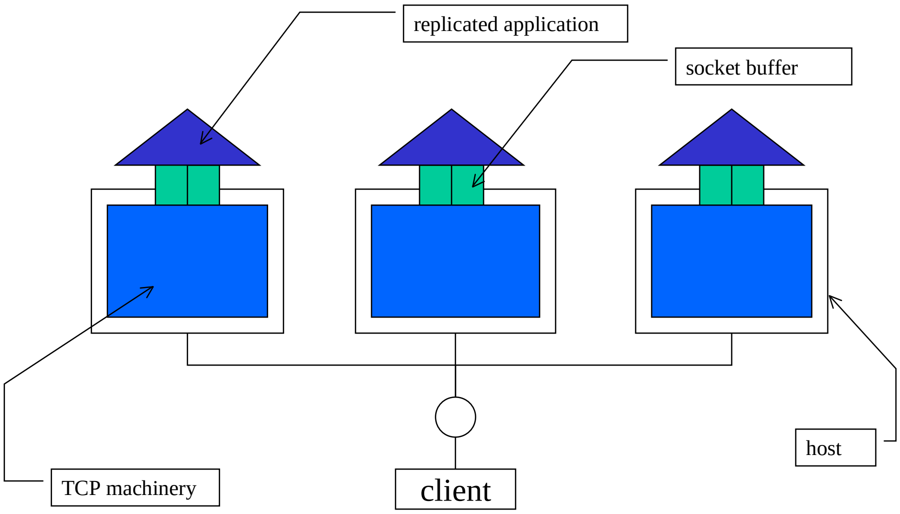
The HYDRANET-FT Approach to Support of Highly-Tolerant Services



HYDRANET-FT: Supporting Tolerant Services through Fault-Tolerant TCP

- Synchronization paradigm
- Atomicity
- Message ordering
- Low-latency error detection
- Virtual synchrony

HYDRANET-FT: Mechanisms



replicated application

socket buffer

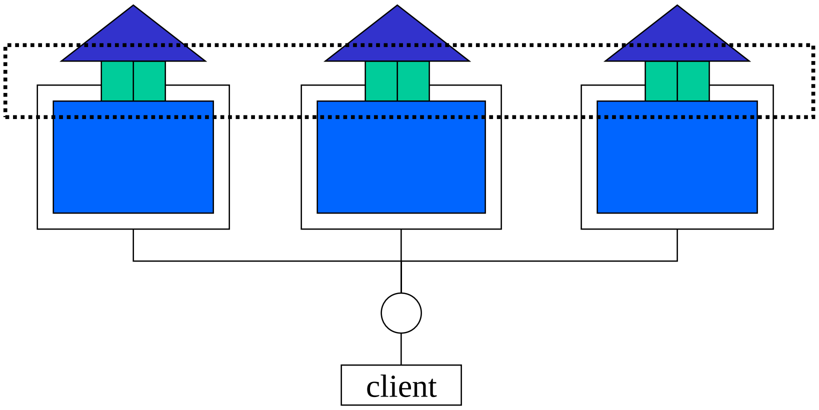
TCP machinery

client

host

HYDRANET-FT: Mechanisms

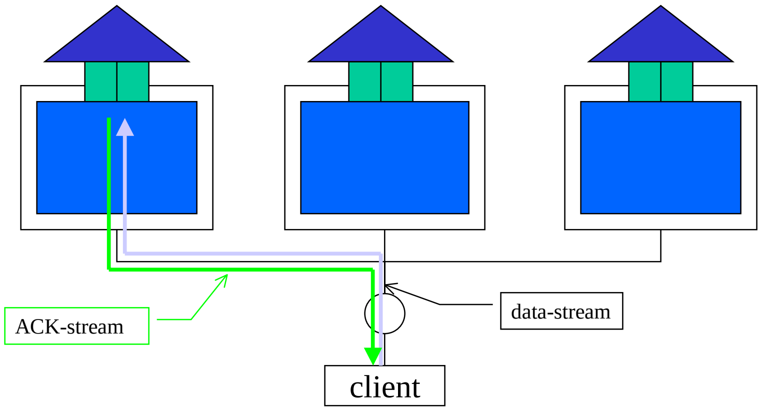
Synchronization at Socket Buffer Layer



client

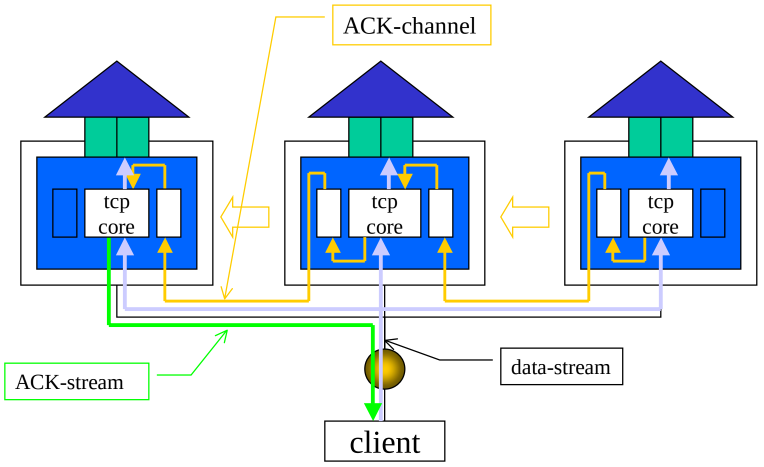
HYDRANET-FT: Mechanisms

Atomicity through Acknowledgement Channel



HYDRANET-FT: Mechanisms

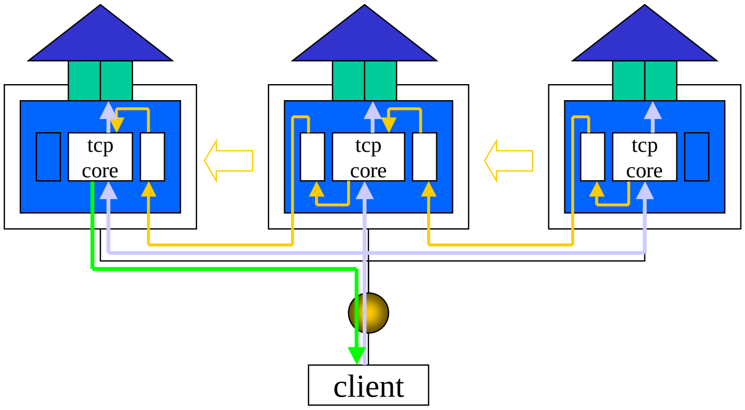
Atomicity through Acknowledgement Channel



HYDRANET-FT: Mechanisms

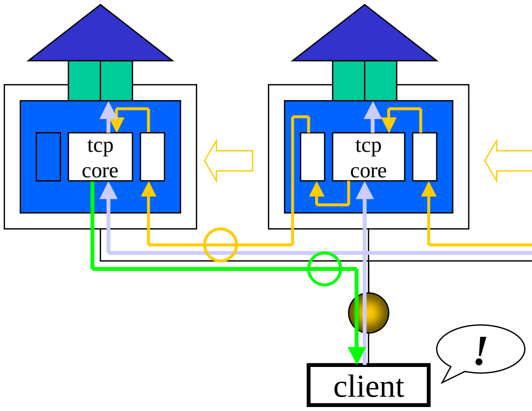
Message Ordering:

- Source ordering enforced by TCP.
- Total ordering across connections enforced by ordered delivery on acknowledgement channel.



HYDRANET-FT: Mechanisms

Failure detection provided by TCP re-transmission at client.

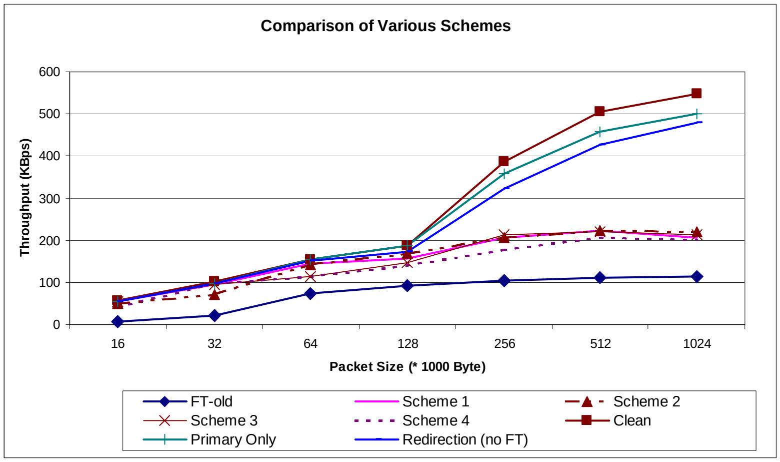


Implementation

- FreeBSD 3.1
- Modifications
  - Process control
  - IP forwarding
  - TCP machinery
- Additional control system calls

```
int v_host(u_long ip_address);
int setportopt(port, mode, detector_parameters)
```

Performance Measurements



Demonstration: Real time Video Streaming

File Type: Mpeg (5 MB)

Frame Rate: 30.2 fps

# Frames: 1509

Player: mTv (for FreeBSD)

Client Host: Dell Inspiron (PII)

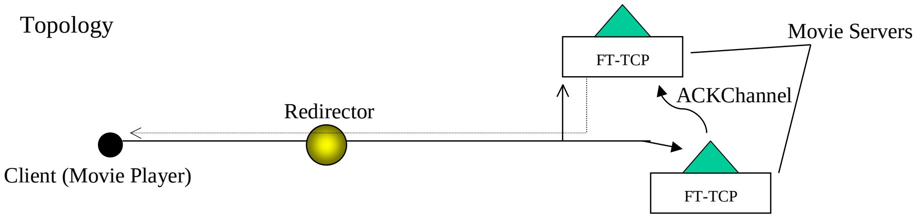
Server Host(s): Intel PI 120 MHz

Router: Intel 486/DX

Network: 10 MBps Ethernet

Infrastructure: HydraNet-FT

Topology



Conclusion

- Summary
  - Starting point: Traditional support for tolerance allocated either too high or too low in protocol stacks to allow for easy deployment on a large scale.
  - HYDRANET-FT
  - Satisfy declared requirements
  - Very-low latency recovery (client invisible)
  - Transparent on *server* side.
- Outlook
  - Integrated control interface for server management
  - Demonstration of behavior for time-critical applications (video streaming)
  - Support for server camouflaging (hiding servers by geographically spreading profile)