Peer-to-Peer Caching Schemes to Address Flash Crowds

T. Stading, P. Maniatis, and M. Baker

Presentation by:

David Hadaller
School of Computer Science
University of Waterloo

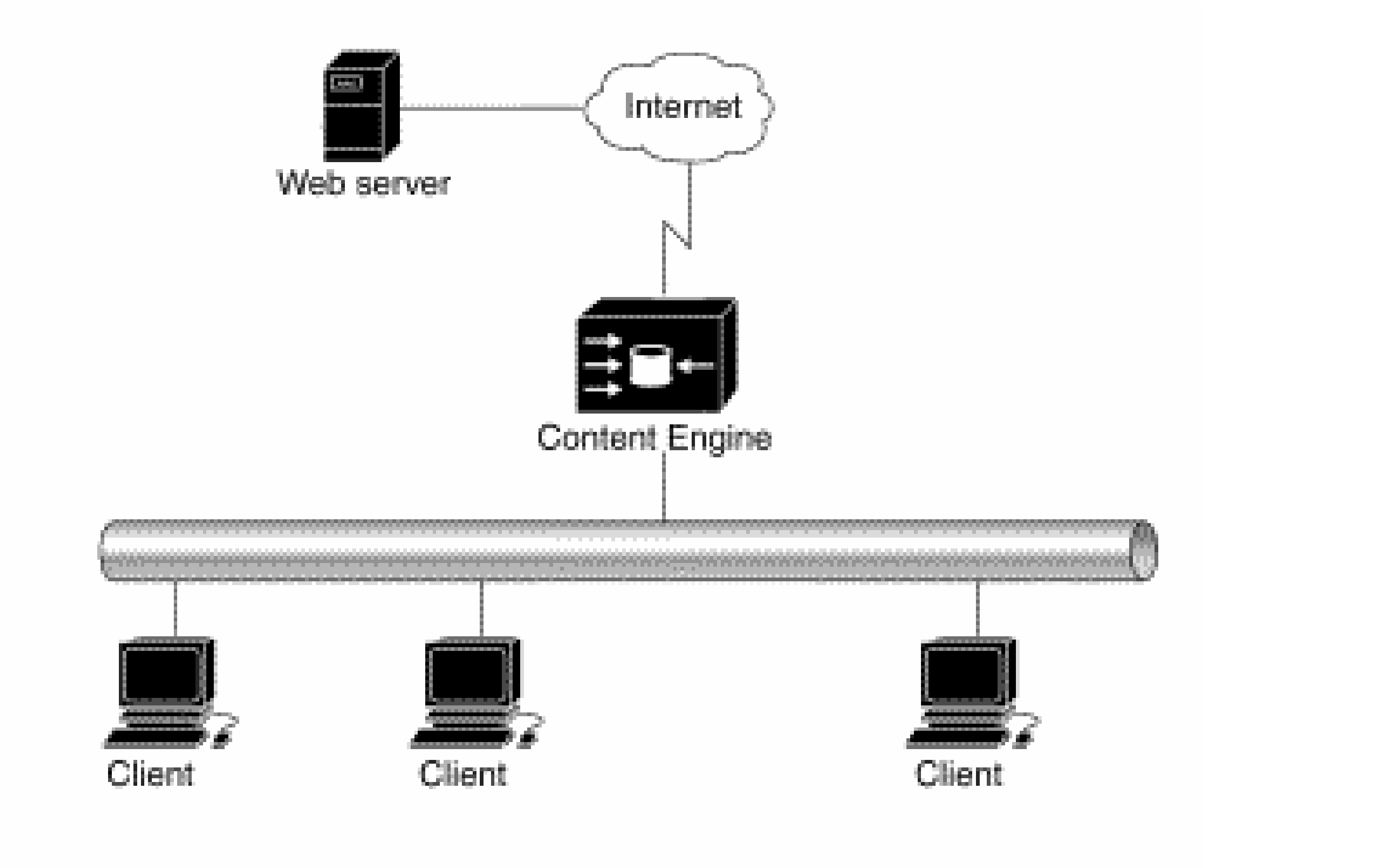
Outline

- Goals of Backslash
- Existing Approaches
- How it Works
- Evaluation
- Status
- Alternatives

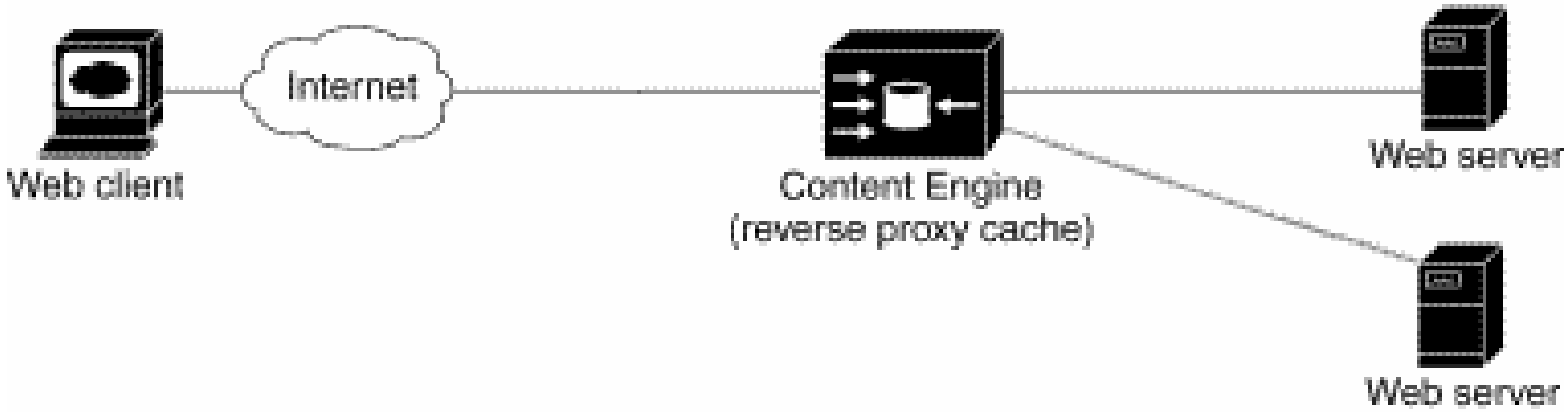
Goals

- Mitigating effects of flash crowds
  - ‘Slashdot’ effect
- Targeted at collaborative organizations
  - Not competing businesses
  - No notion of fairness or pricing

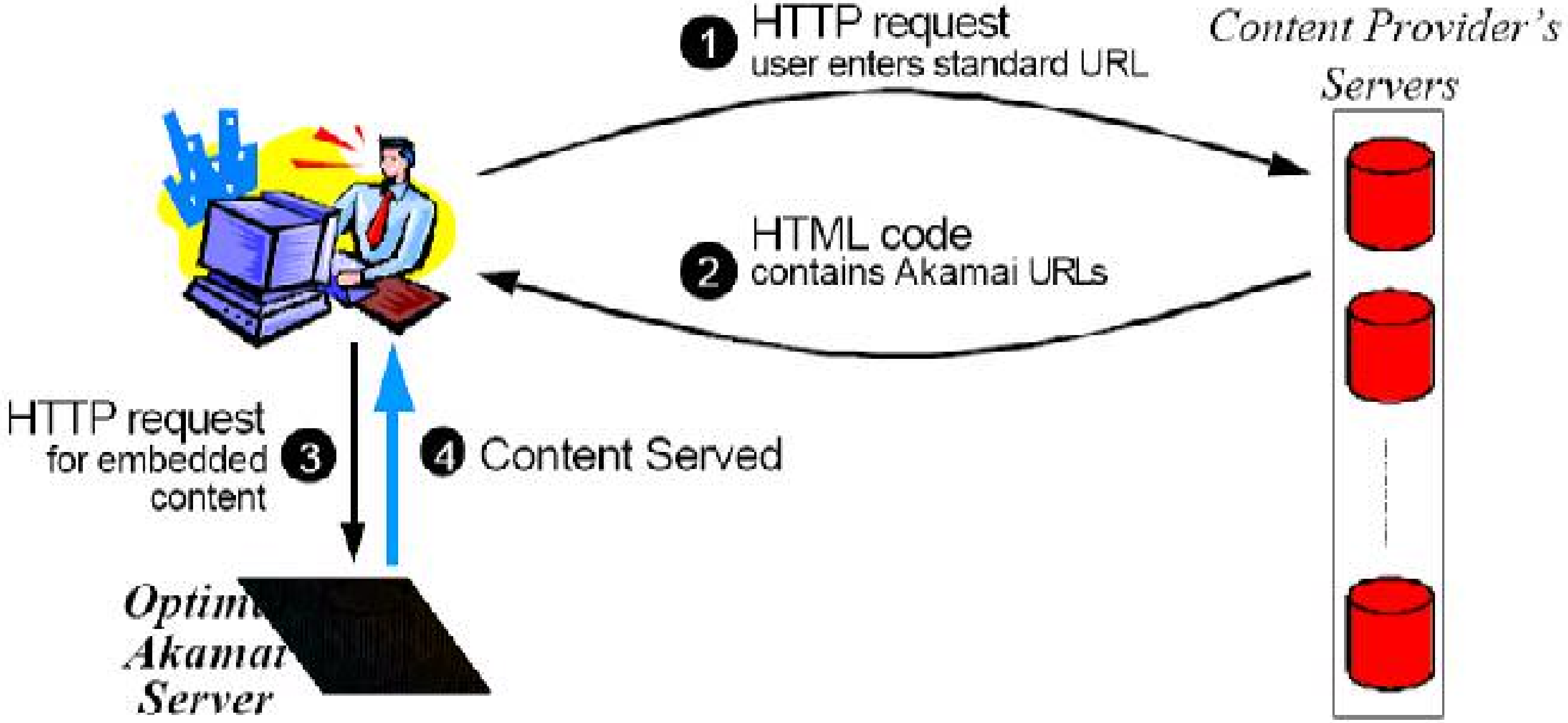
Forward Proxy Cache



Reverse Proxy Cache



Akamai


1 HTTP request
user enters standard URL
Content Provider's Servers
2 HTML code
contains Akamai URLs
HTTP request for embedded content 3
4 Content Served
Optim
Akamai
Server

Goals

- These require hardware investment
- Backslash: same service, no cost

How it Works

- If one web server exceeds load threshold, offload some requests to others
- Rewriting and Redirecting
  - Using DNS and HTTP tricks

How it Works

- URL Rewriting, DNS

- http://www.backslash.stanford.edu/image.jpg
- is replaced with
- http://<hash>.backslash.berkeley.edu/www.backslash.stanford.edu/image.jpg

- backslash.berkeley.edu performs lookup in DHT and resolves to the IP address of a node containing desired document

How it Works

- URL Rewriting, HTTP

- http://www.backslash.stanford.edu/image.jpg
- is replaced with
- http://a.b.c.d/www.backslash.stanford.edu/image.jpg

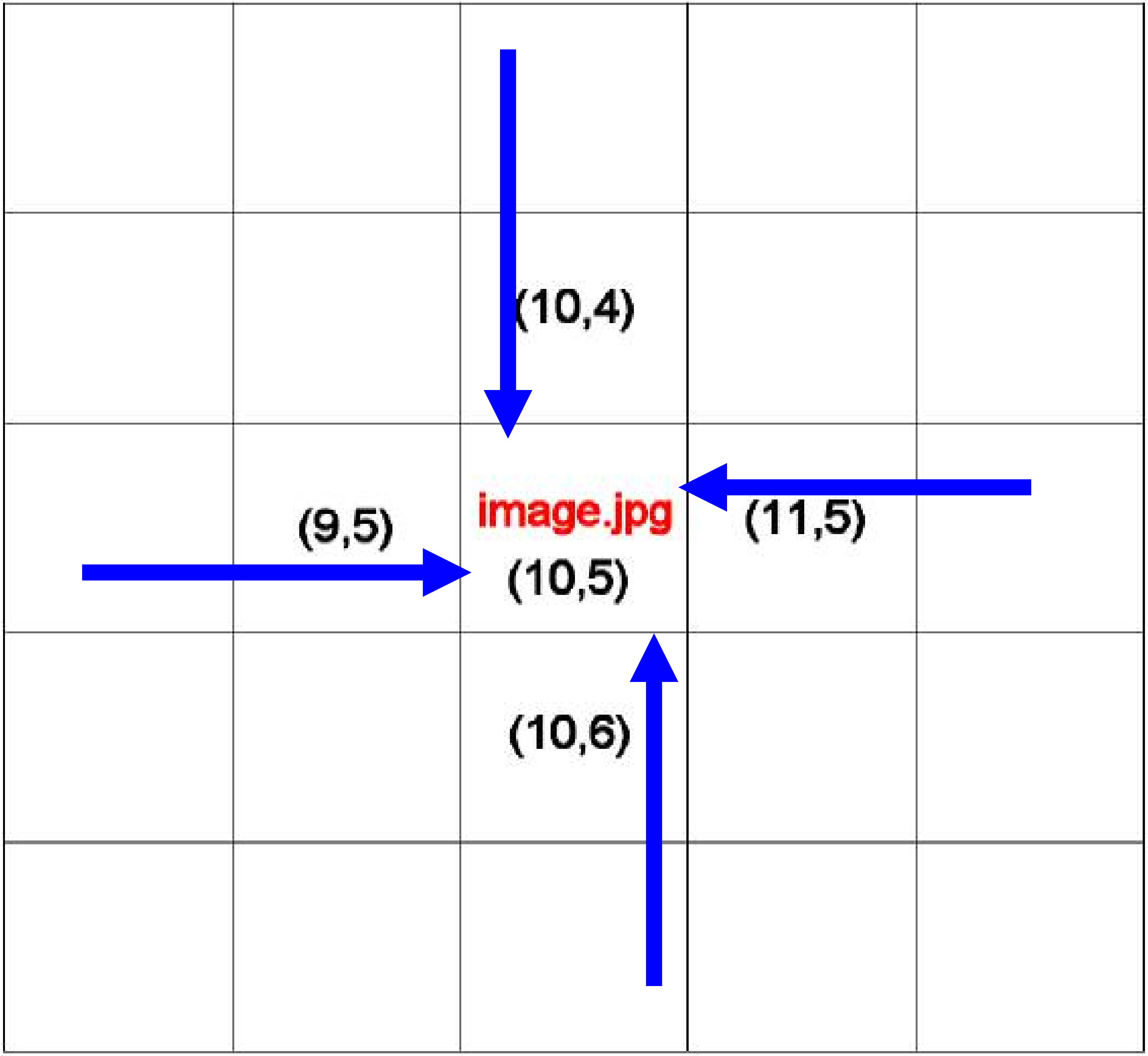
- a.b.c.d is the IP address of a node containing the desired document
- DHT lookup is done by overloaded node

How it Works

- DNS redirect
  - www.stanford.edu resolves to IP of surrogate server
- HTTP redirect
  - Web server responds with REDIRECT message

Replication

- Local diffusion
  - Push threshold
    - Based on a counter (not rate)
  - Replicate one overlay hop closer to source
    - Bubble effect in CAN


(10,4)
(9,5)
image.jpg
(10,5)
(11,5)
(10,6)

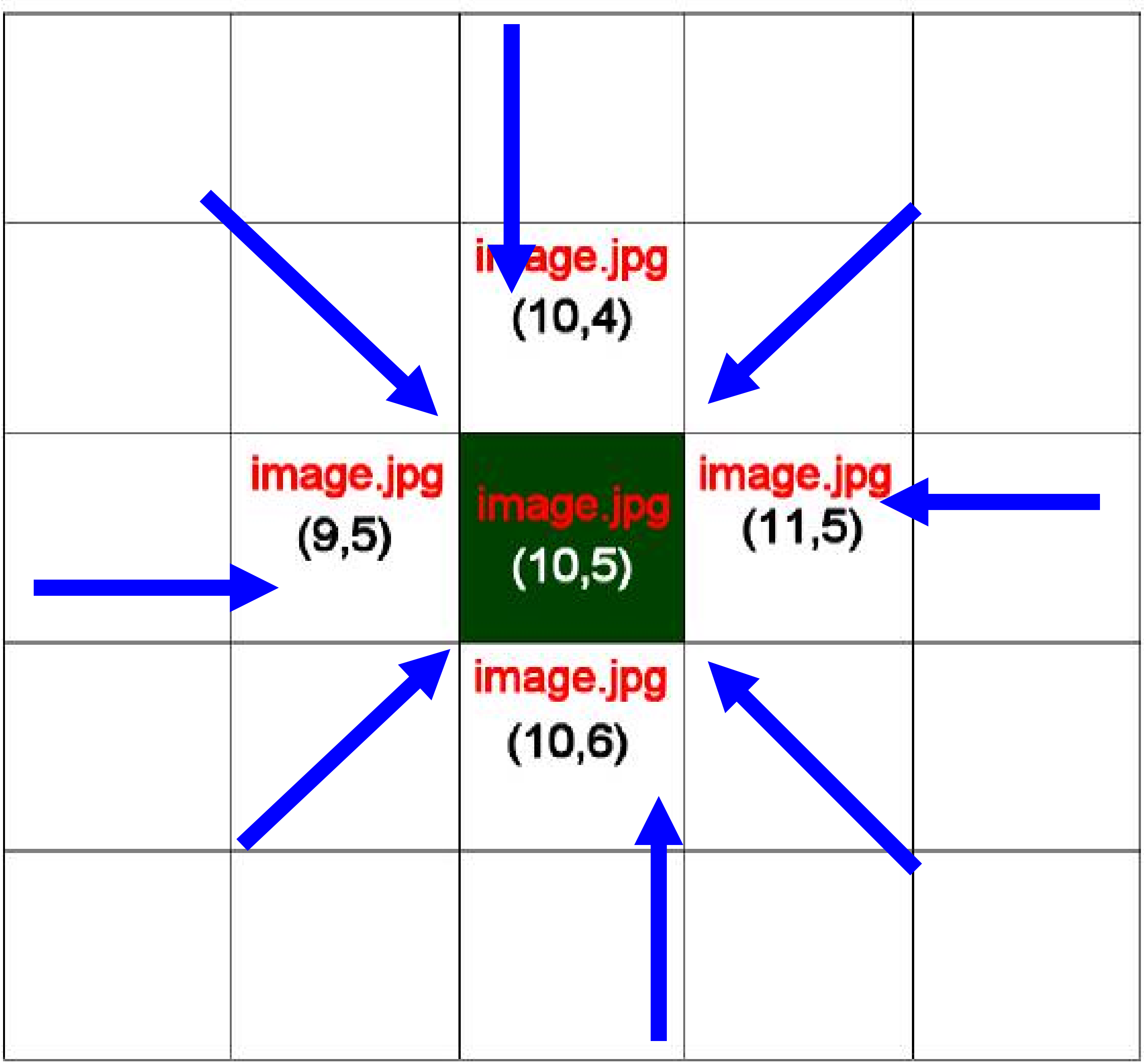


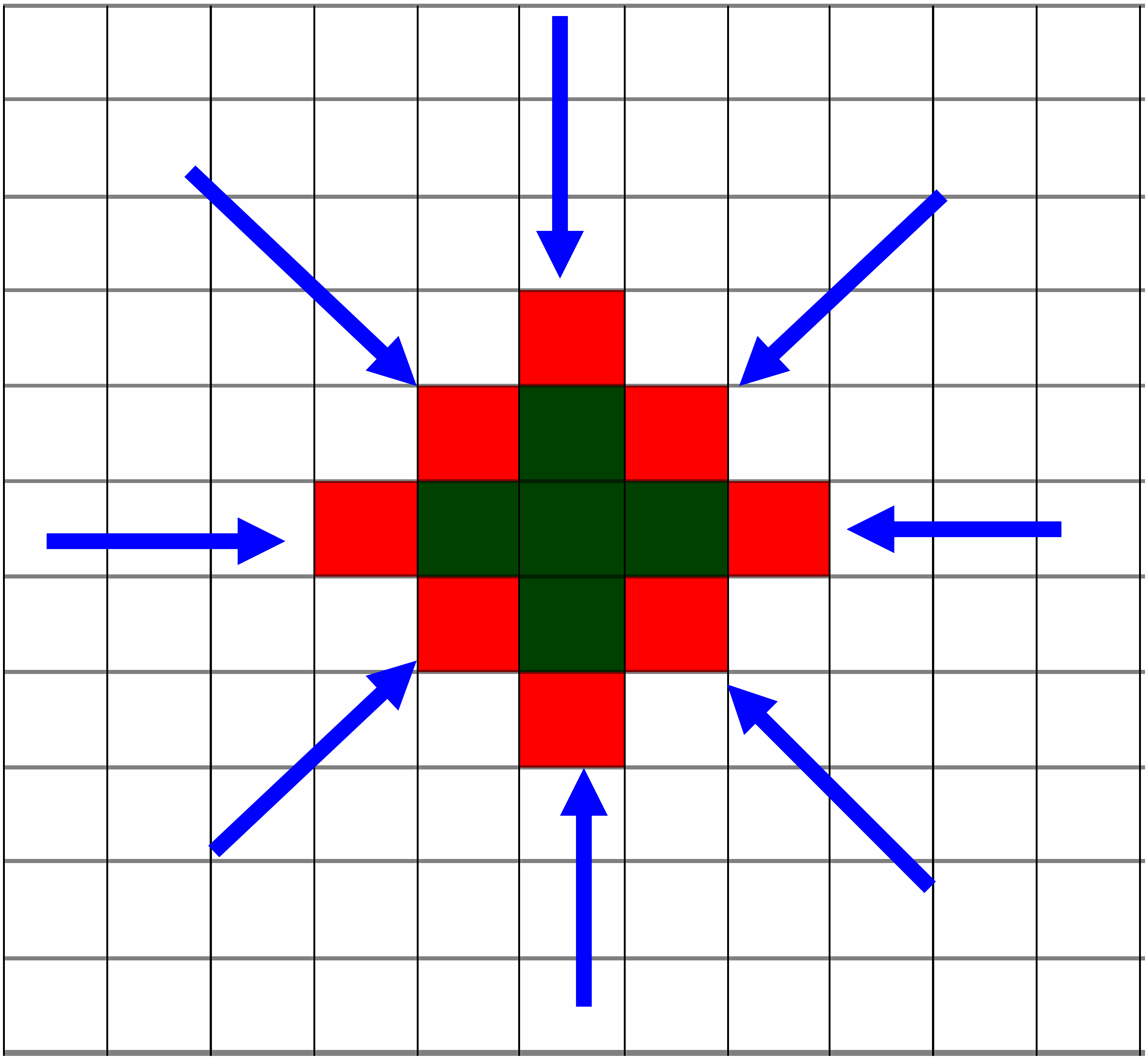
image.jpg (10,4)

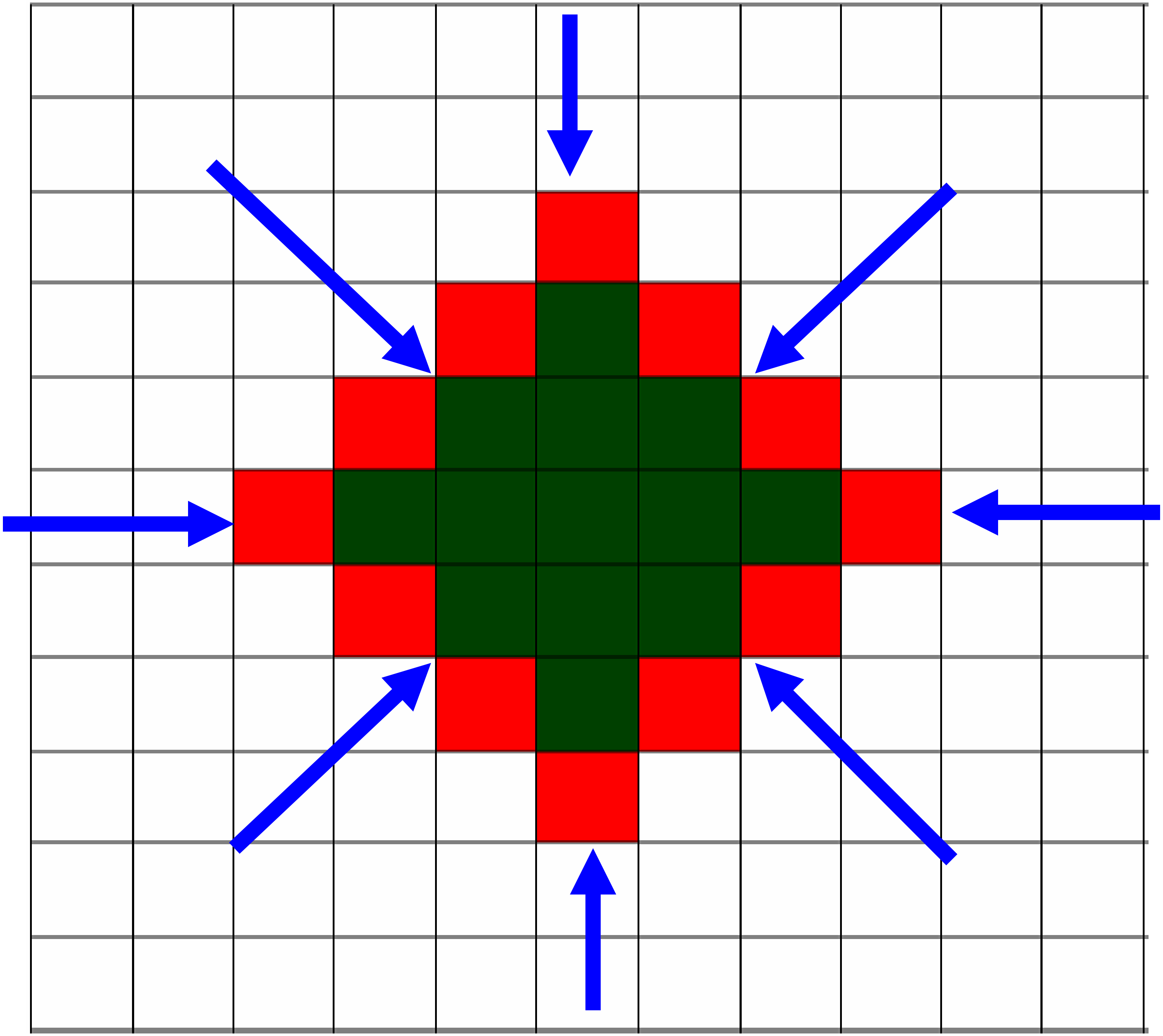
image.jpg (9,5)

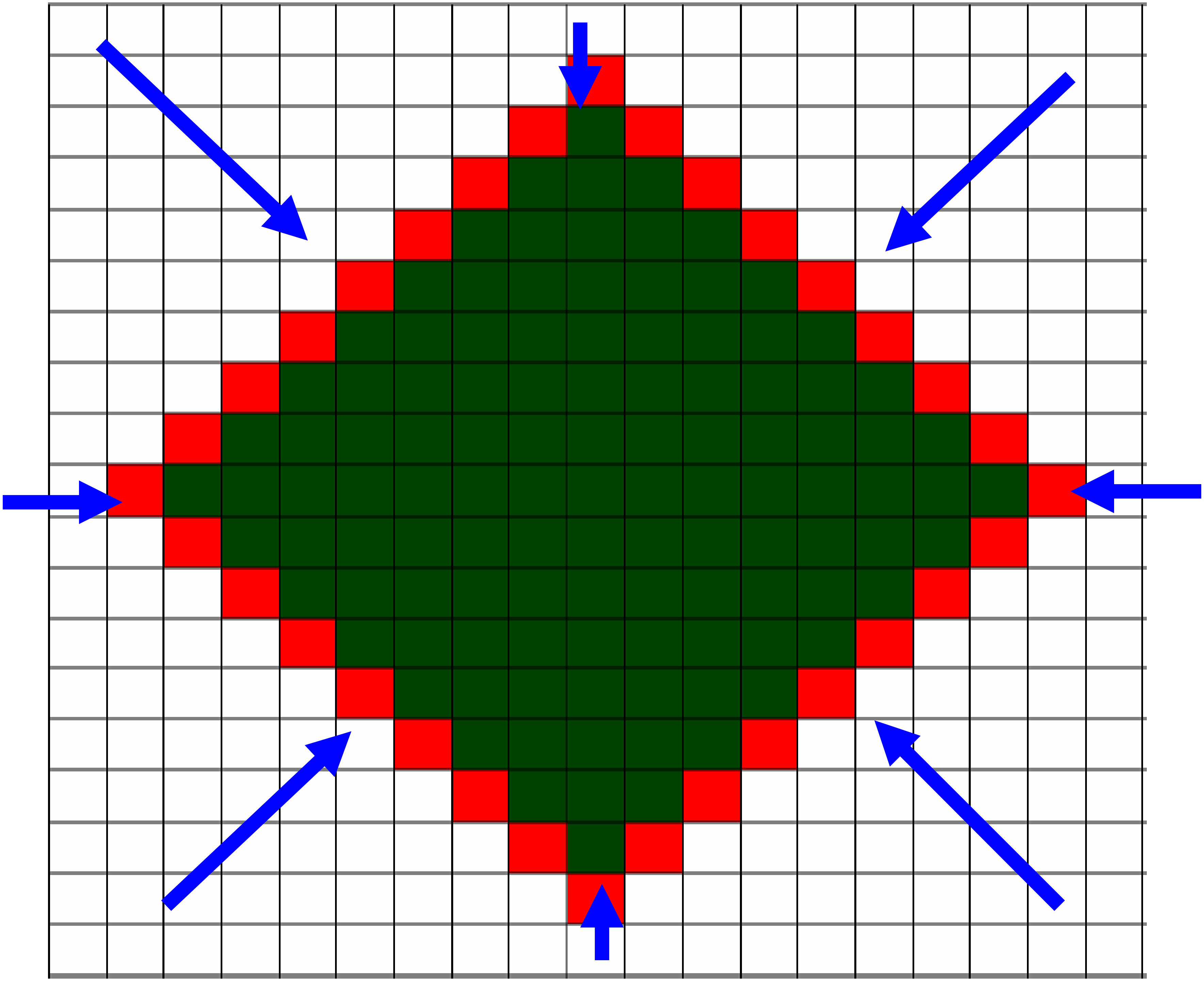
image.jpg (10,5)

image.jpg (11,5)

image.jpg (10,6)



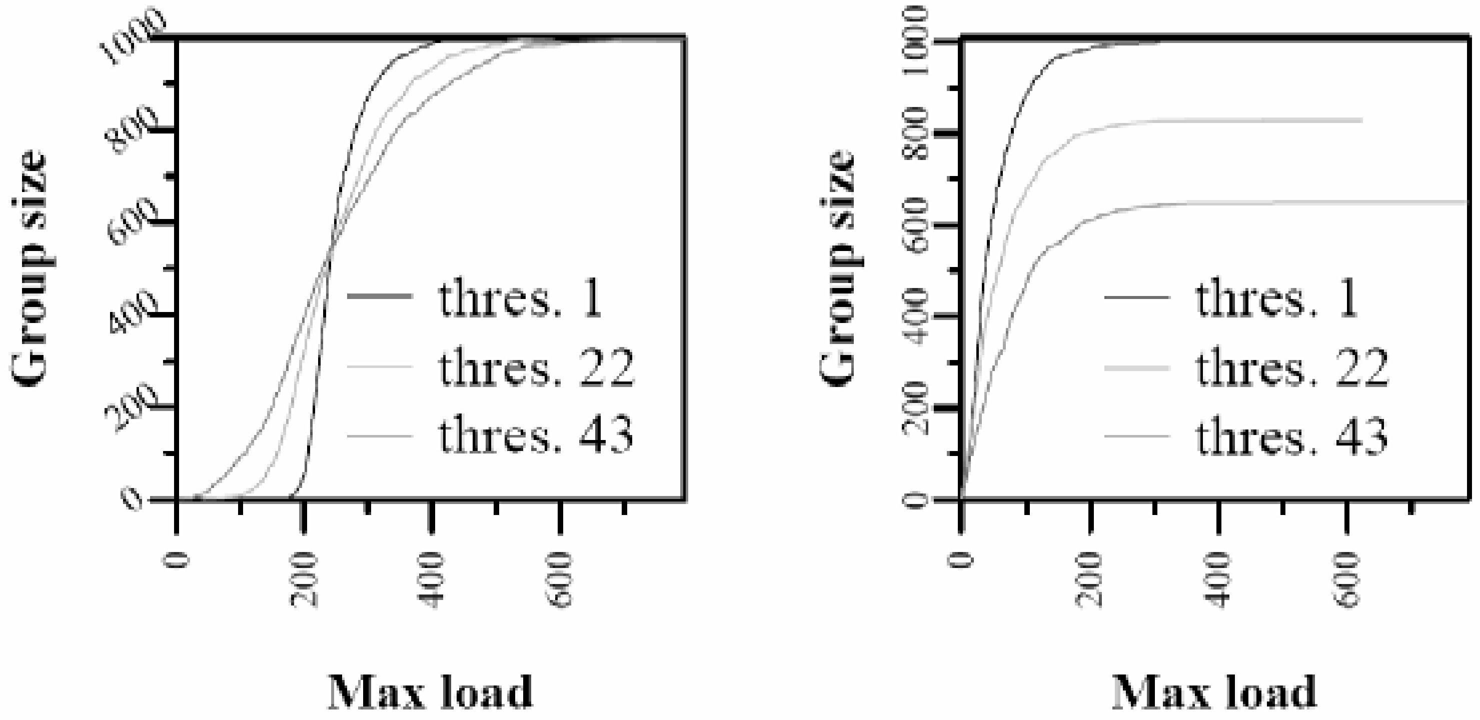




Replication

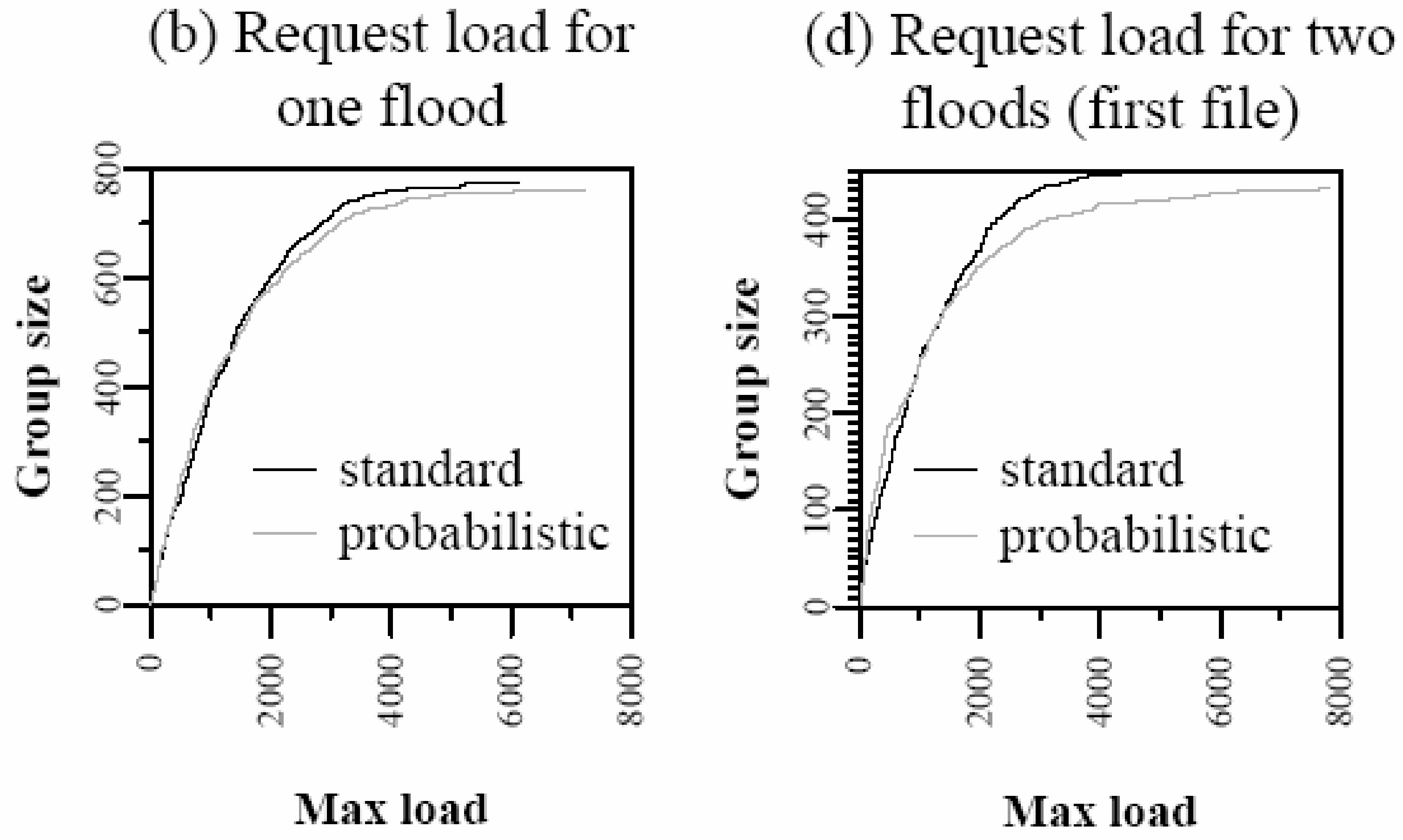
- Directory diffusion
  - Pointers to documents
  - Return list of document locations to requestor
- Probabilistic forwarding
  - Idea: reduce load on perimeter nodes of bubble
  - Linearly decreasing forwarding probability

Evaluation


(c) Request load of first file

(d) Request load of second file

Evaluation

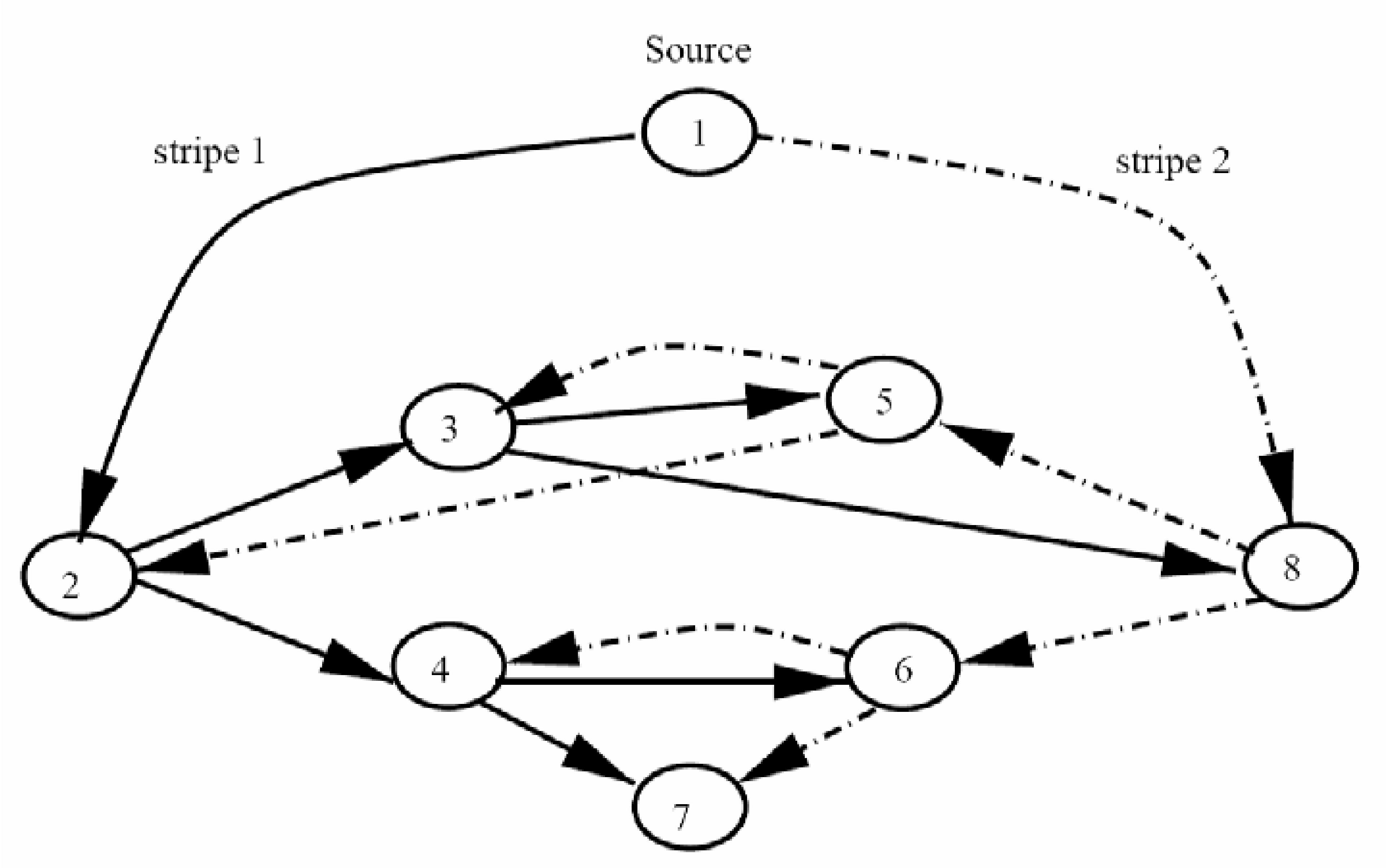


(b) Request load for one flood

(d) Request load for two floods (first file)

Alternatives

- Client-side caching
  - Squirrel: Use DHT to index files on clients
- Cooperative Parallel-Download Systems
  - Multicasting
    - SplitStream: Use DHT to distribute file stripes, using erasure codes. Tree topology determines multicast path.
    - CoopNet: similar idea but uses central coordinator instead of DHT



SplitStream

Conclusions

- Many promises to extend this preliminary work
  - Has not yet been followed up
- Main contribution
  - transparent server-side solution
- Comments