

Sistemas Distribuídos

José Orlando Pereira

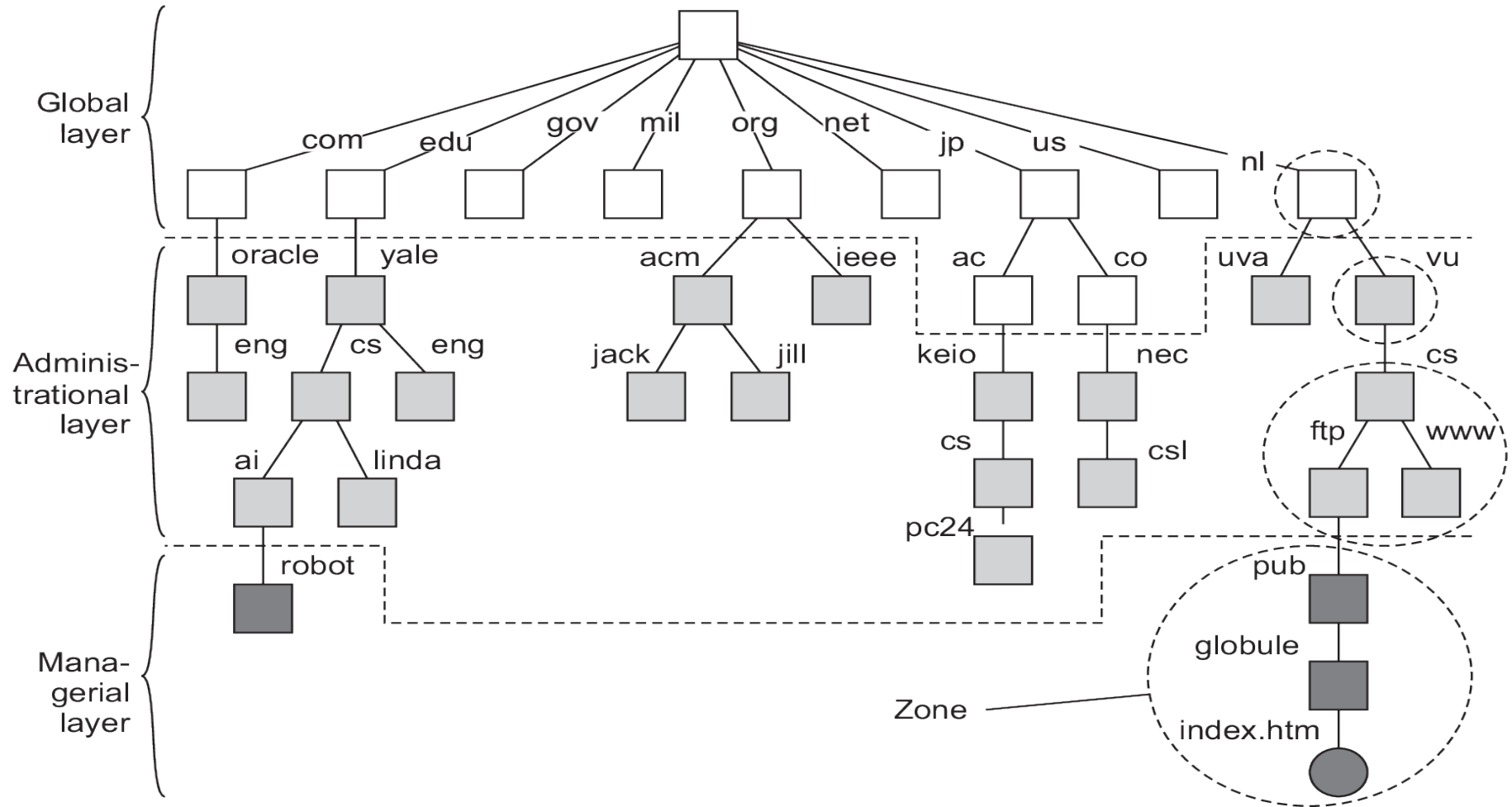
Departamento de Informática
Universidade do Minho



Decentralized systems

- Distributed systems with:
 - Very large number of participants
 - Dynamic participation (churn)
 - Geographically distributed
 - Multiple authority and administration domains
- Typical problems:
 - Lookup by name (e.g., file sharing)
 - Information dissemination (e.g., event notification)
- A.k.a. peer-to-peer or P2P

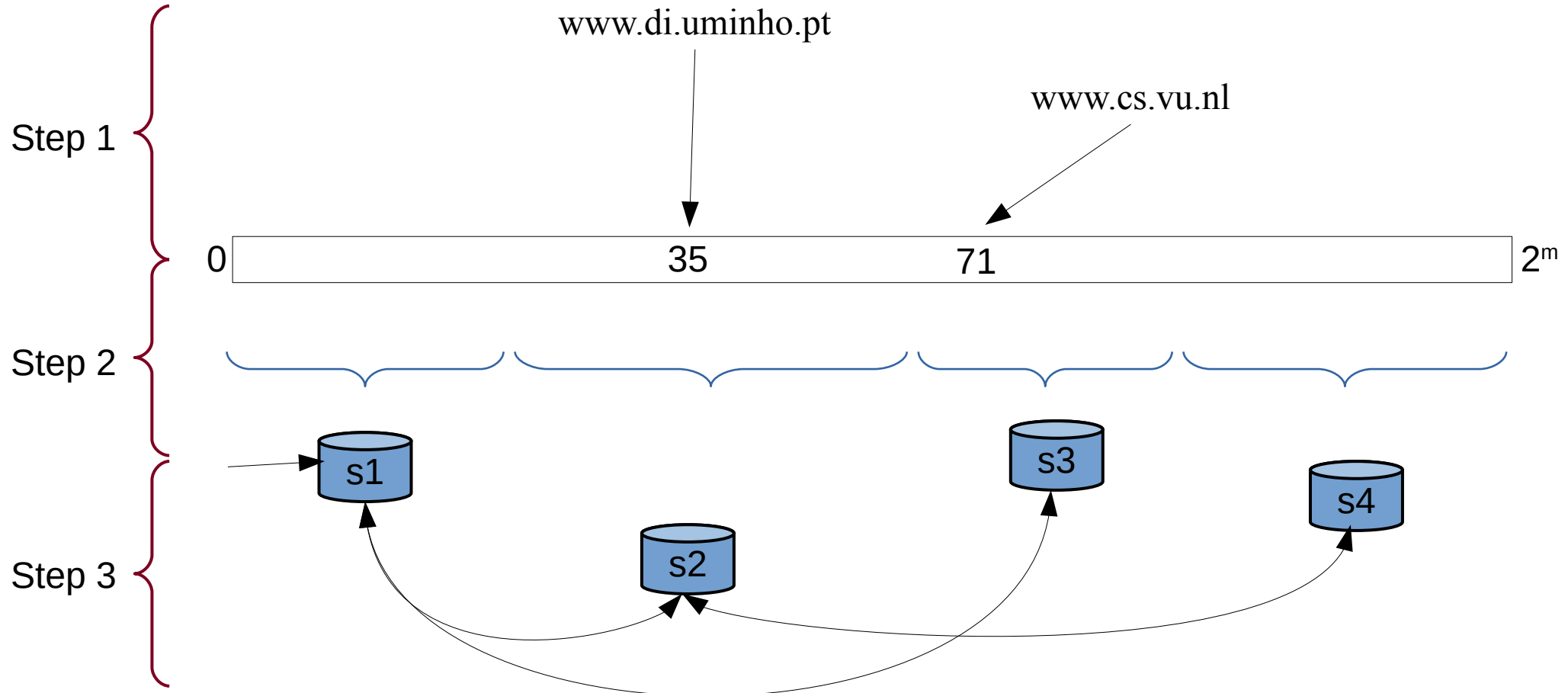
Hierarchical lookup



Hierarchical lookup

- Efficient lookup:
 - Depth of tree and number of hops is $\sim \log N$
- Allows distributed but coordinated administrative authority
- There is still a bottleneck and SPOF at root node

Distributed Hash Table (DHT)

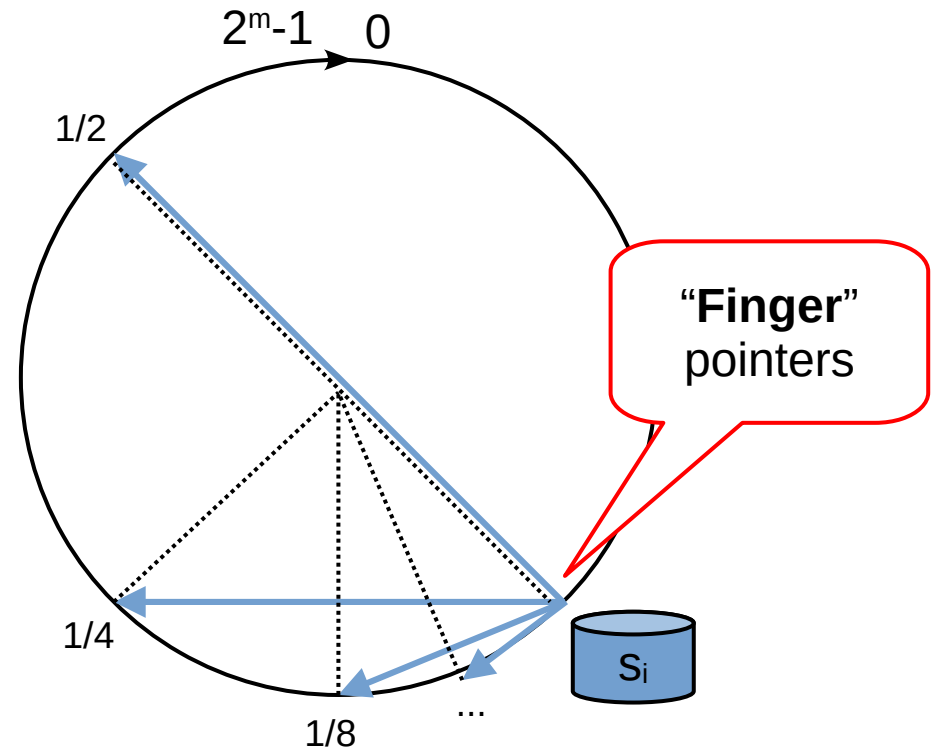


Distributed Hash Table (DHT)

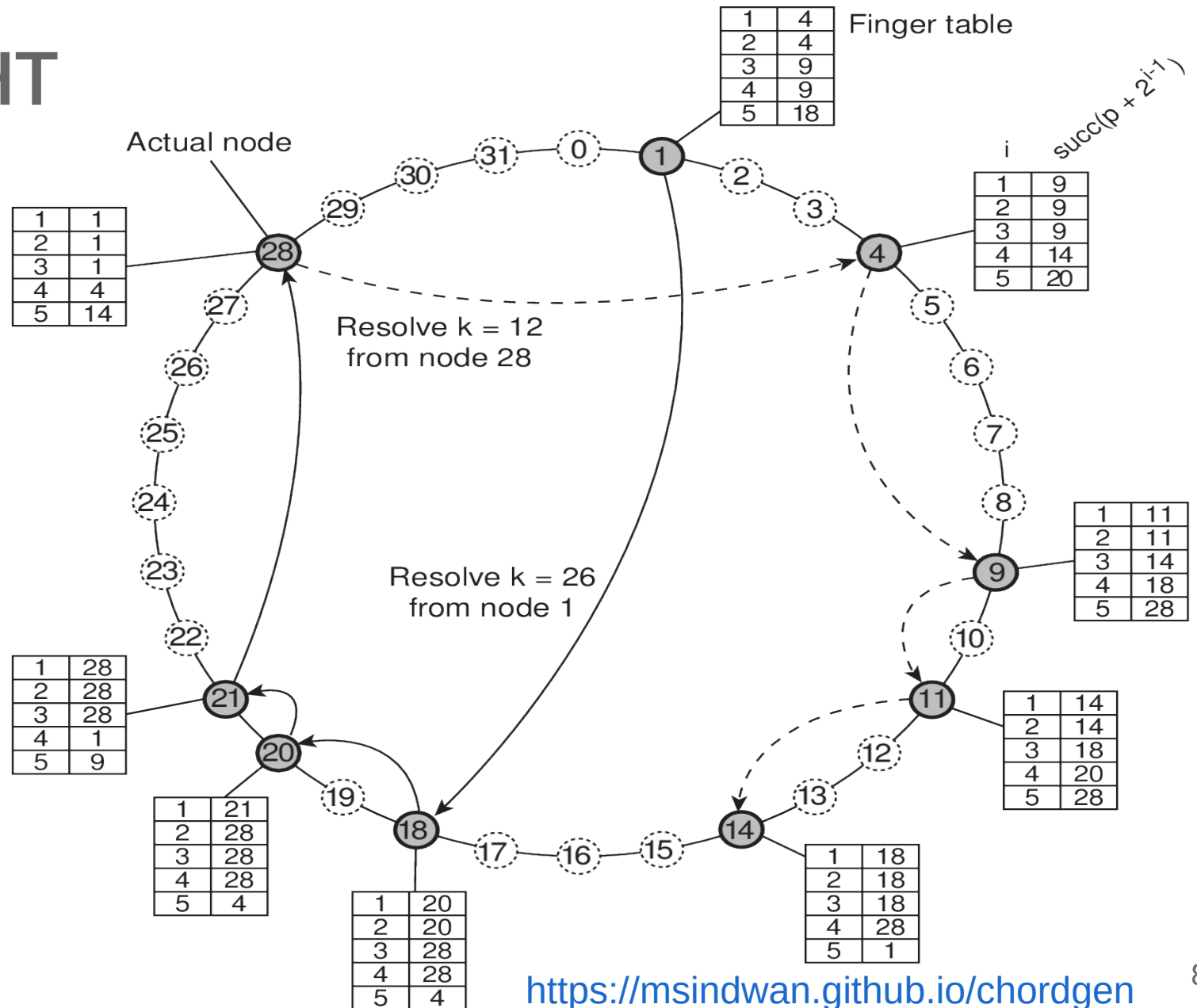
- Steps in distributed hashing:
 - 1) map names to integers in an interval (bucket)
 - 2) map integer intervals (buckets) to servers
 - 3) build overlay network that finds the desired servers
- Within each server, use any data structure to map names to objects

Chord DHT

- Wrap around hash space
- Split in halves:
 - At most m times
- Results in m pointers in each node
- Lookup: $O(m)$

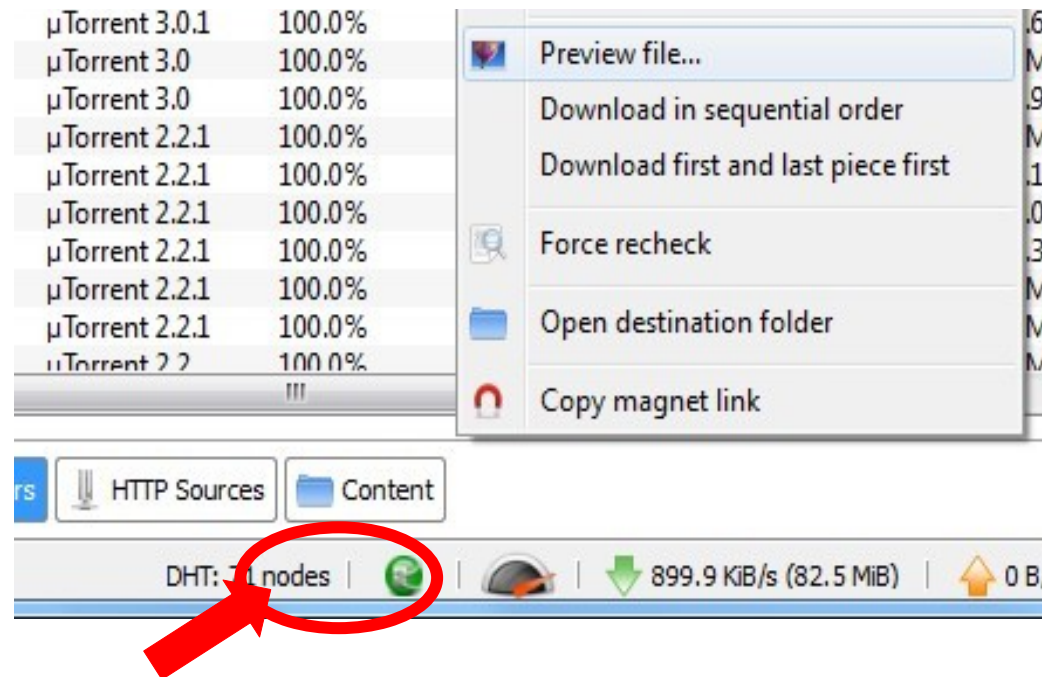


Chord DHT



Distributed Hash Table (DHT)

- No single root node:
 - No bottleneck and no SPOF
- Efficient lookup:
 - $\sim \log N$ hops
- Example: BitTorrent

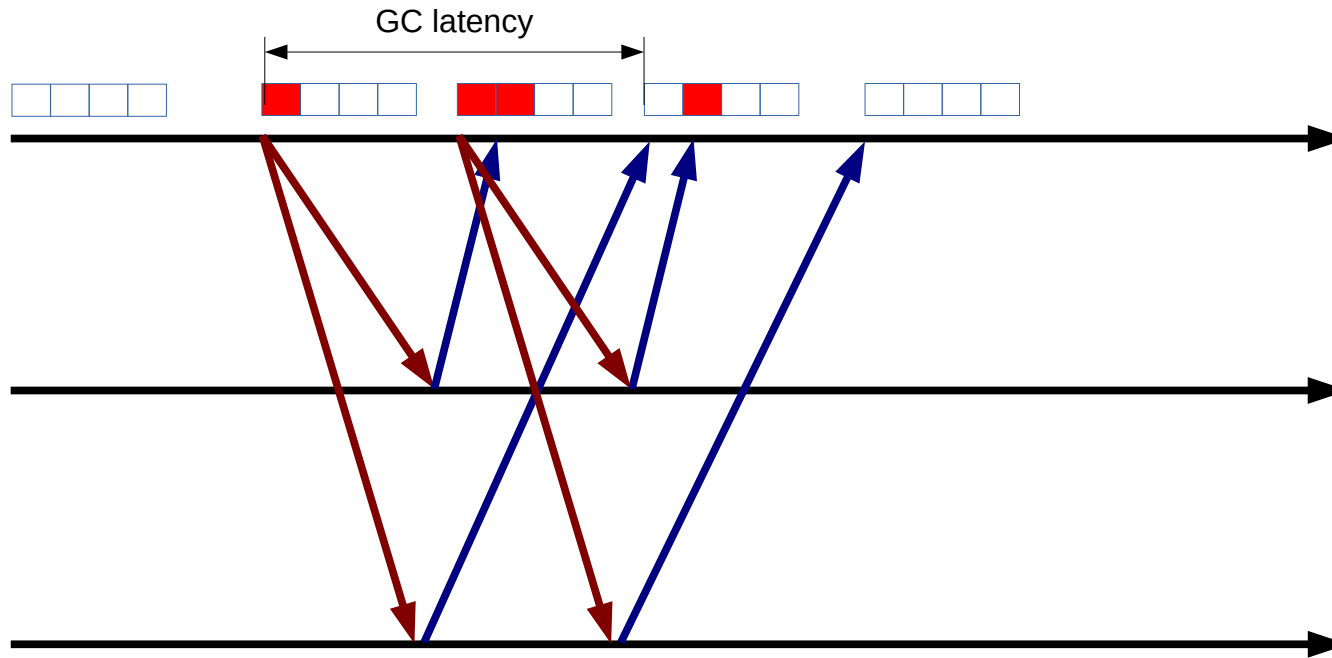


Application Level Multicast

- Reliably send to multiple destinations (group)
- Informally, reliability means that:
 - all destinations deliver all messages sent
- In reality, senders and receiver fail, thus:
 - all correct destinations deliver the same messages
- Worst case scenario: Message to only some of the destinations

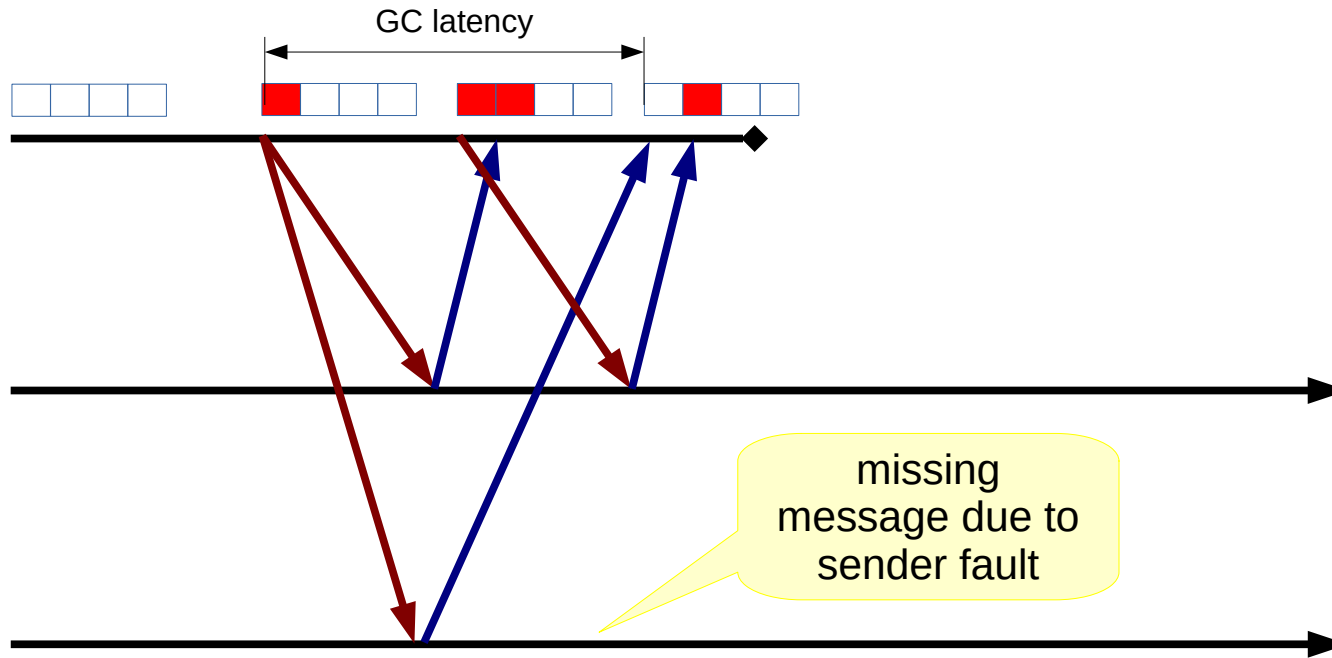
General approach

- Buffer and retransmit until acknowledged



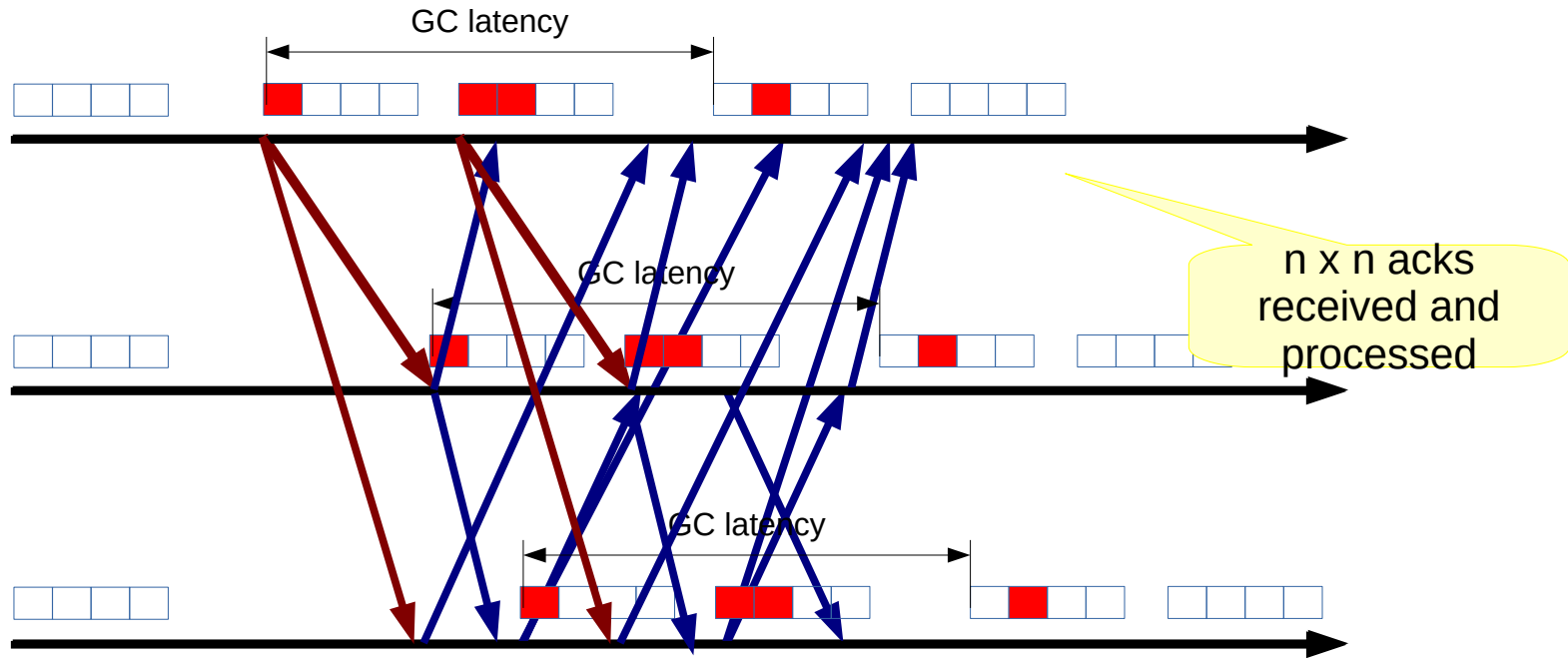
General approach

- Depends on sender correctness:



Multicast with agreement

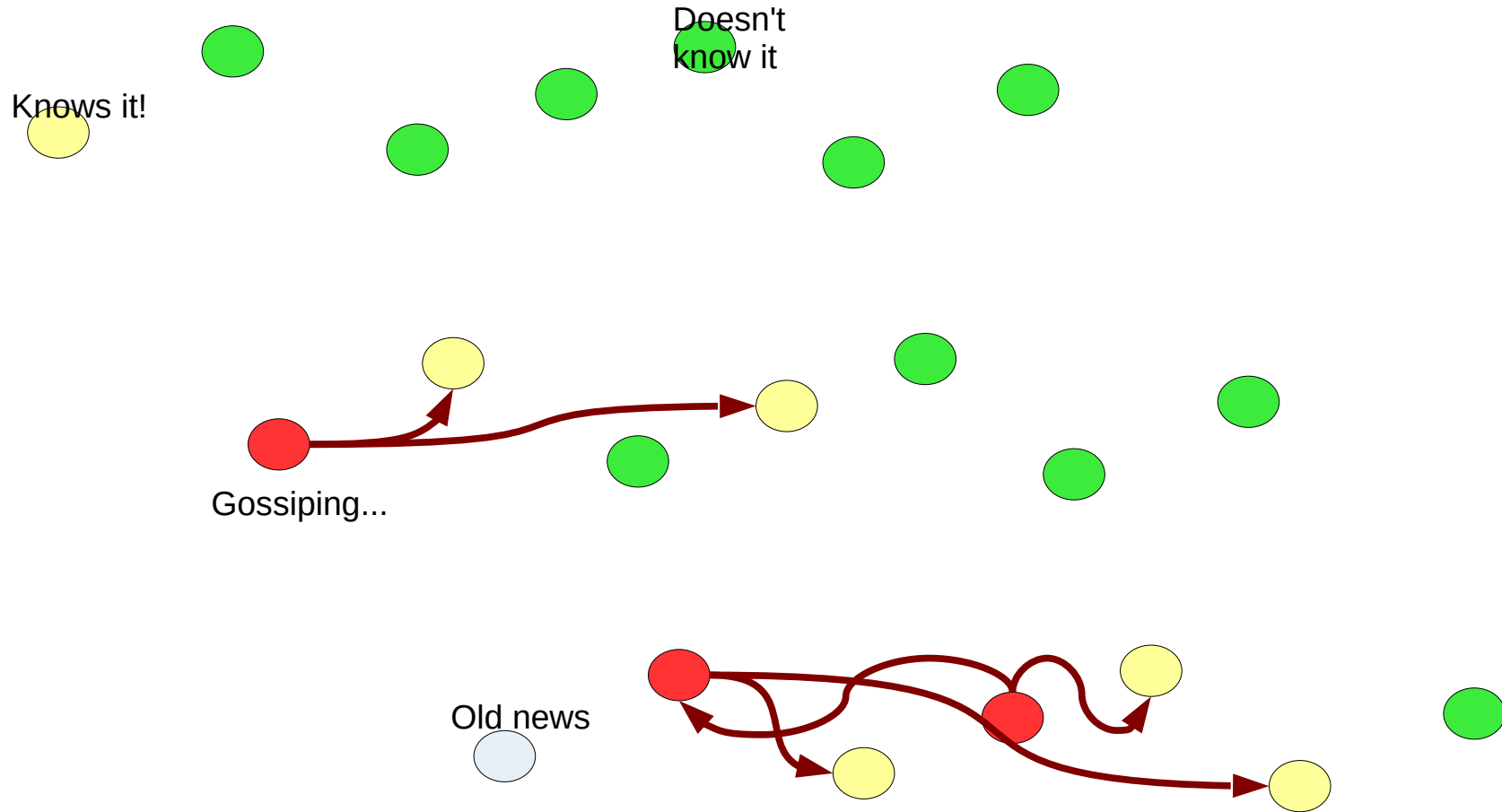
- Acks need to be sent to all destinations resulting in “ $O(n^2)$ ack implosion”:



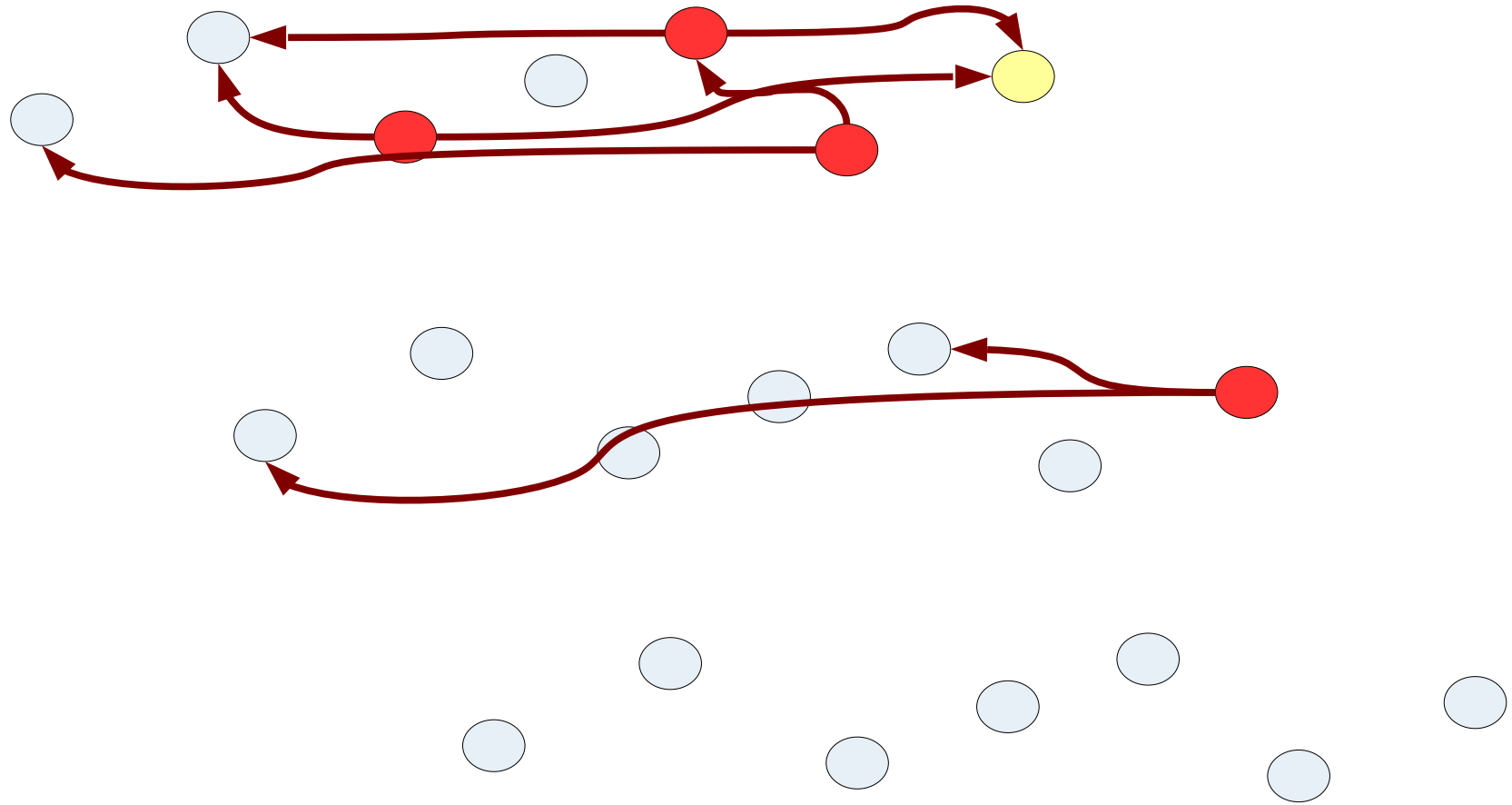
Gossip

- Simple protocol to multicast a message:
 - Select a small subset of random targets
 - Forward message only to those targets
 - Discard message
- Upon receiving a new message, act as the sender

Gossip



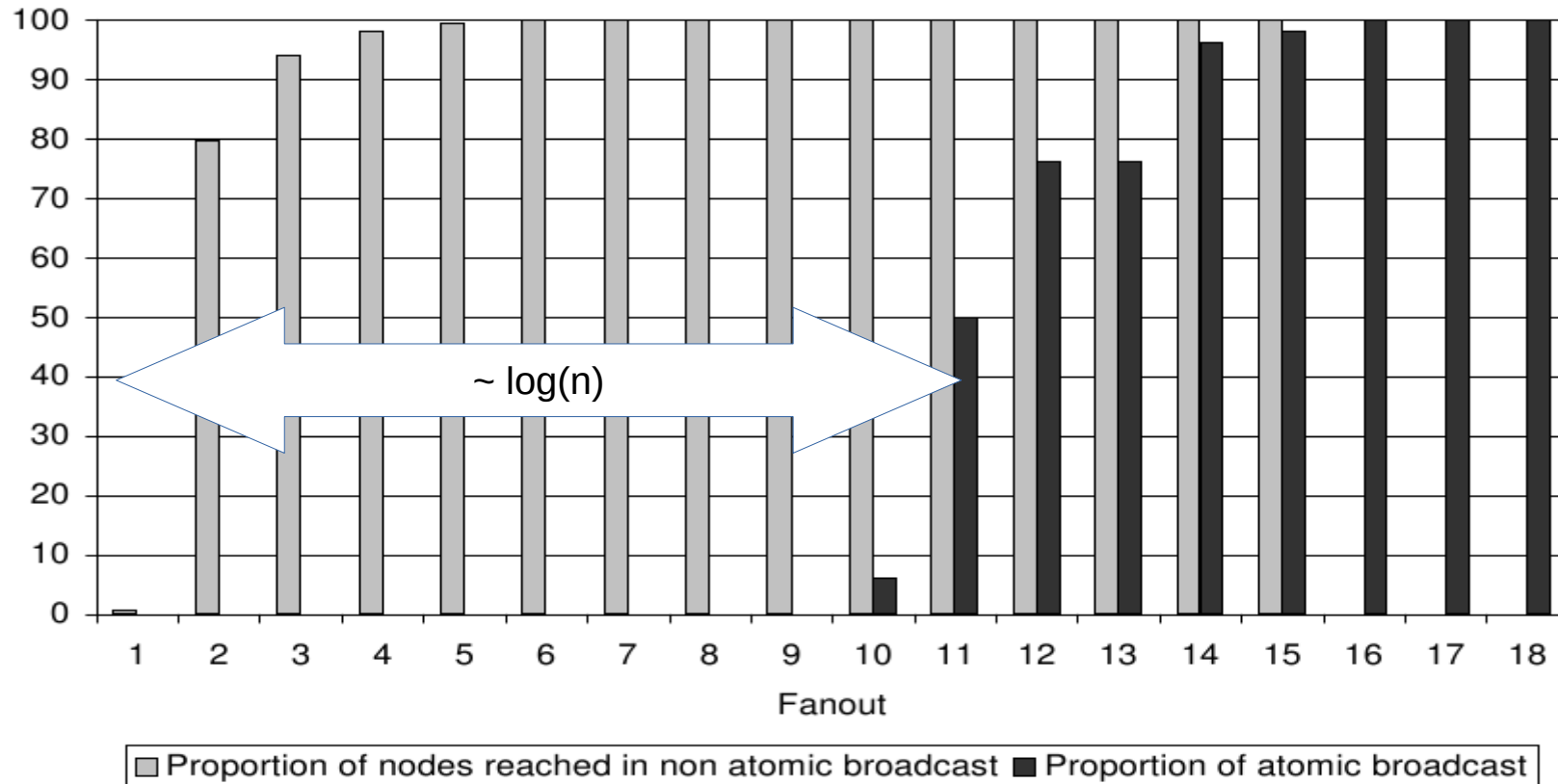
Gossip



Gossip and Epidemics

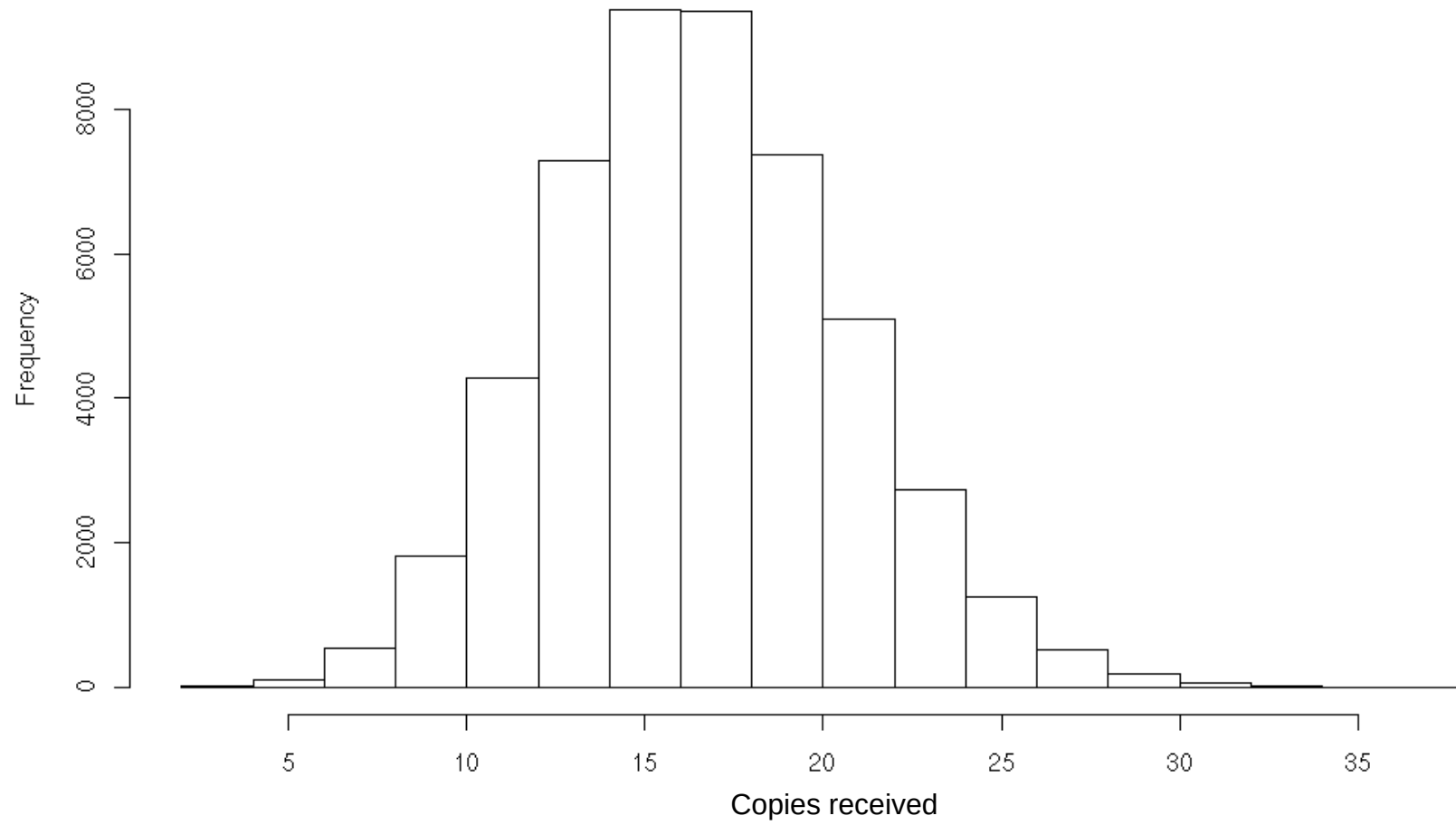
- Similarity with epidemics:
 - Sender = contagious = spreads rumor
 - Receiver = infected = knows rumor
 - Ignores duplicated = dead = old news...
- Interesting parameters:
 - n – size of the population
 - f – number of targets

Fanout vs Reliability



(50000 destinations)

Redundancy



Summary

- DHTs are now the scalable, dependable, widely used solution to distributed search
- Gossip protocols implicitly ensure w.h.p. that all correct processes have received the same messages