~

BY LEITÃO, PEREIRA, AND RODRIGUES

HYPARVIEW: A MEMBERSHIP PROTOCOL FOR RELIABLE GOSSIP-BASED BROADCAST

IT SEEMS THAT DISTRIBUTED COMPUTATIONS ARE INTRINSICALLY DIFFICULT TO UNDERSTAND, AND PERHAPS A SIMPLE WAY TO DESCRIBE THEIR BEHAVIOR DOES NOT EVEN EXIST.

Schwarz and Mattern

GOSSIP PROTOCOLS ARE AN INTERESTING APPROACH BECAUSE THEY ARE HIGHLY RESILIENT . . .

GOSSIP MADE OBVIOUS

- Broadcasting or Replicating Information
- Prefer TCP over UDP for sharing information
- Messages can be tailored to “at least once” delivery

THE AIM OF A MEMBERSHIP SERVICE IS TO MAINTAIN THESE PARTIAL VIEWS SATISFYING A NUMBER OF GOOD PROPERTIES.

PROPERTIES OF A GOOD PARTIAL VIEW MANAGER

- Connectivity: Nodes don’t remain isolated
- Degree Distribution: No particular nodes are too popular or important
- Average Path Length: Propagating a message should be efficient
- Clustering Coefficient: Minimize waste within neighbors
- Accuracy: views contain up-to-date information

REACTIVE VIA SCAMP

- Places special value on JOIN and LEAVE messages to update views
- Failure detectors are critical
- Caveat: Scamp actually uses leases and heartbeats

SCAMP: Peer-to-peer lightweight membership service for large-scale group communication

Ayalvadi J. Ganesh, Anne-Marie Kermarrec and Laurent Massoulié

Microsoft Research Ltd., 7JJ Thomson Avenue Cambridge CB3 0FB, UK
{ajg,annemk,lmassoul}@microsoft.com

Abstract. Gossip-based protocols have received considerable attention for broadcast applications due to their attractive scalability and reliability properties. The reliability of probabilistic gossip schemes studied so far depends on each user having knowledge of the global membership and choosing gossip targets uniformly at random. The requirement of global knowledge is undesirable in large-scale distributed systems.

In this paper, we present a novel peer-to-peer membership service which operates in a completely decentralized manner in that nobody has global knowledge of membership. However, membership information is replicated robustly enough to support gossip with high reliability. Our scheme is completely self-organizing in the sense that the size of local views naturally converges to the 'right' value for gossip to succeed. This 'right' value is a function of system size, but is achieved without any node having to know the system size. We present the design, theoretical analysis and preliminary evaluation of SCAMP. Simulations show that its performance is comparable to that of previous schemes which use global knowledge of membership at each node.

Keywords scalability, reliability, peer-to-peer, gossip-based probabilistic multicast, membership, group communication, random graphs.

CYCLIC VIA CYCLON

- Periodically shuffle view with a neighbor
- Generally, the periodicity is large relative to the traffic of regular messages

BOTH REACTIVE AND CYCLIC

- JOIN, NEIGHBOR, DISCONNECT are reactive messages
- SHUFFLE and SHUFFLEREPLY are cyclic
- Active View: small number of connections
- Passive View: larger number of candidates
- Active Views are symmetric

MAINTAINING THE ACTIVE VIEW

- JOIN requests will always put the node into the active view of at least the contact
- JOIN may trigger a DISCONNECT with an active node
- Disconnected nodes are moved into the passive view
- FORWARDJOIN ensures that the node is subsequently placed in other nodes' active and passive views
- NEIGHBOR requests are used to promote nodes from the passive view to the active view

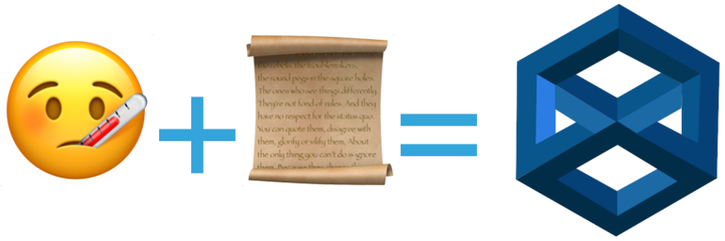
MAINTAINING THE PASSIVE VIEW

- Periodically send a SHUFFLE request
- SHUFFLE requests get forwarded with a TTL
- SHUFFLEREPLY includes just passive nodes from a particular node

MASSIVE PERCENTAGES OF FAILURES HAVE ALMOST NO VISIBLE IMPACT ON HYPARVIEW...

MEIKLEJOKE'S ON YOU

PARTISAN: A BRIEF HISTORY



Partisan

build failing

Partisan is a flexible, TCP-based membership system fo

Partisan features:

- Single node testing, facilitated by a disterl control c
  at.
- Messages are sent via TCP connections that are ma
- Failure detection is performed TCP.
- Connections are verified at each gossip round.
- Configurable fanout.
- On join, gossip is performed immediately, instead of
- HyParView implementation.

Partisan has many available peer service managers:

- Full membership with TCP-based failure detection:
- Client/server topology: `partisan_client_server_pe`
- HyParView, hybrid partial view membership protoco
  `partisan_hyparview_peer_service_manager.`
- Static topology: `partisan_static_peer_service_ma`

OTP Compatibility ✕
#212 opened 4 days ago by cmeiklejohn

Analysis 2 ✕
#211 opened 8 days ago by cmeiklejohn

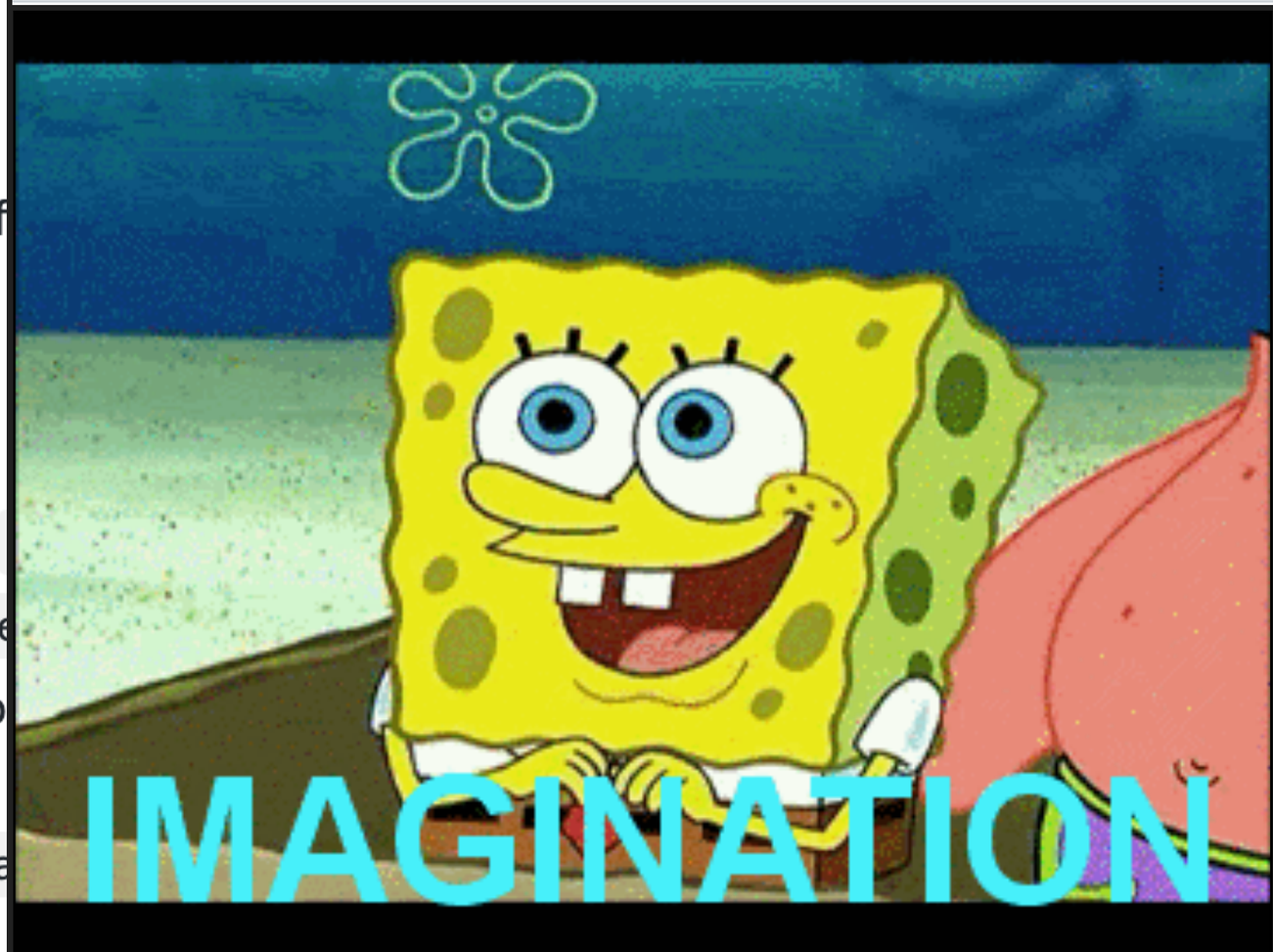
WIP: Lashup ✕
#210 opened 13 days ago by cmeiklejohn

WIP: Paxoid ✓
#204 opened 25 days ago by cmeiklejohn

WIP: Riak Ensemble for Paxos ✕
#202 opened on Mar 18 by cmeiklejohn

WIP: Analysis and Model Checking ✕
#201 opened on Mar 15 by cmeiklejohn

[V4] Peer service refactoring. ✓
#189 opened on Oct 8, 2018 by cmeiklejohn



THANKS FOR LISTENING

@NEILMENNE