Herd Distributed Messaging

Peter Halliday CS5412

May 4, 2012

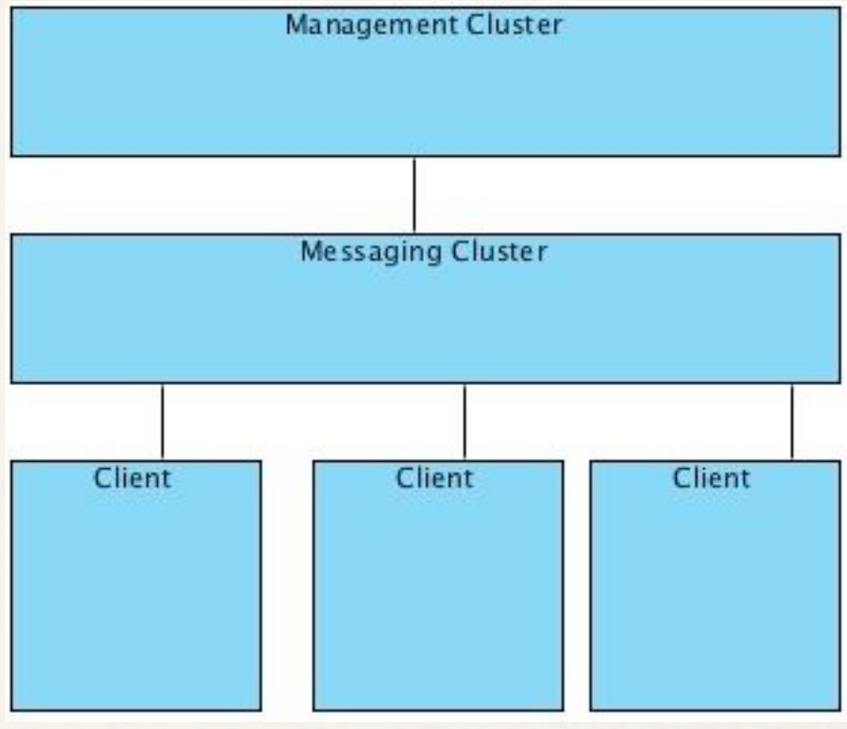
Project Overview

- Peer-to-Peer version of Twitter is really like a distributed group messaging. (Twitter meets IM meets IRC)

- Project Goals:
  - Both management and messaging should be distributed
  - Should allow for as little management as necessary
  - Should allow for failure

Project Architecture

- Herd Management Layer
  - DHT, Python, Gevent
- Herd Message Cluster
  - Gossip, Python, Tornado
- Messaging
  - Using JSON



Design Decisions

- Communication targets are chosen randomly to maximize resilience.
- The Management Cluster and Messaging Cluster both use seeds, but refresh their seeds on the fly.
- The Act of passing message to group doesn't assume there's a local thread in the Management Cluster.

Test Suite

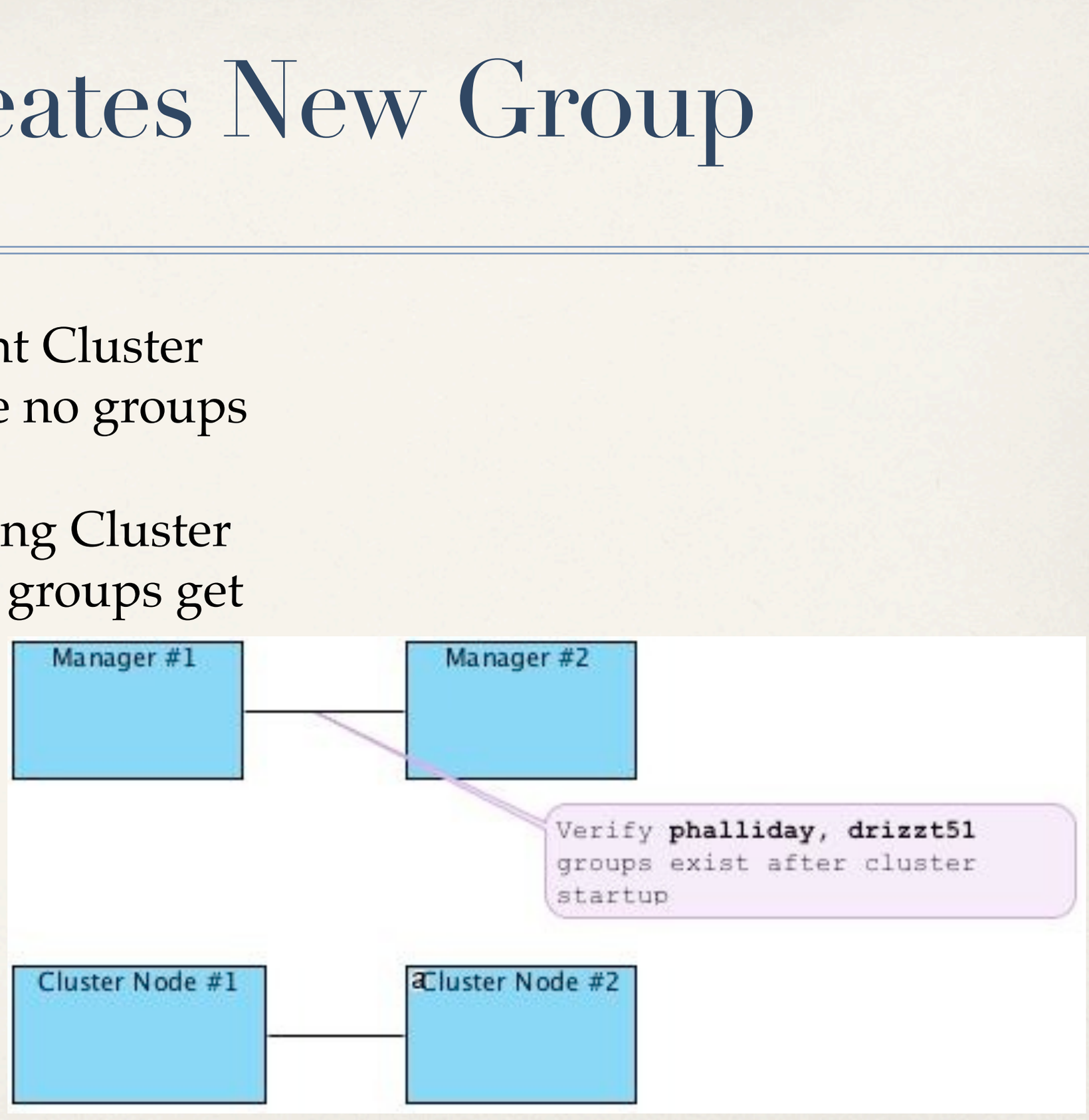
- 632 lines of production code across 7 files
- 552 lines of testing code across 20 tests (All passing)
- Test Coverage at 75%

Demo

- Basic Scenario
  - 2 Management Nodes (127.0.0.1, 127.0.0.4)
  - 2 Message Cluster Nodes (127.0.0.2, 127.0.0.3)
- 5 Common Scenarios

Client Creates New Group

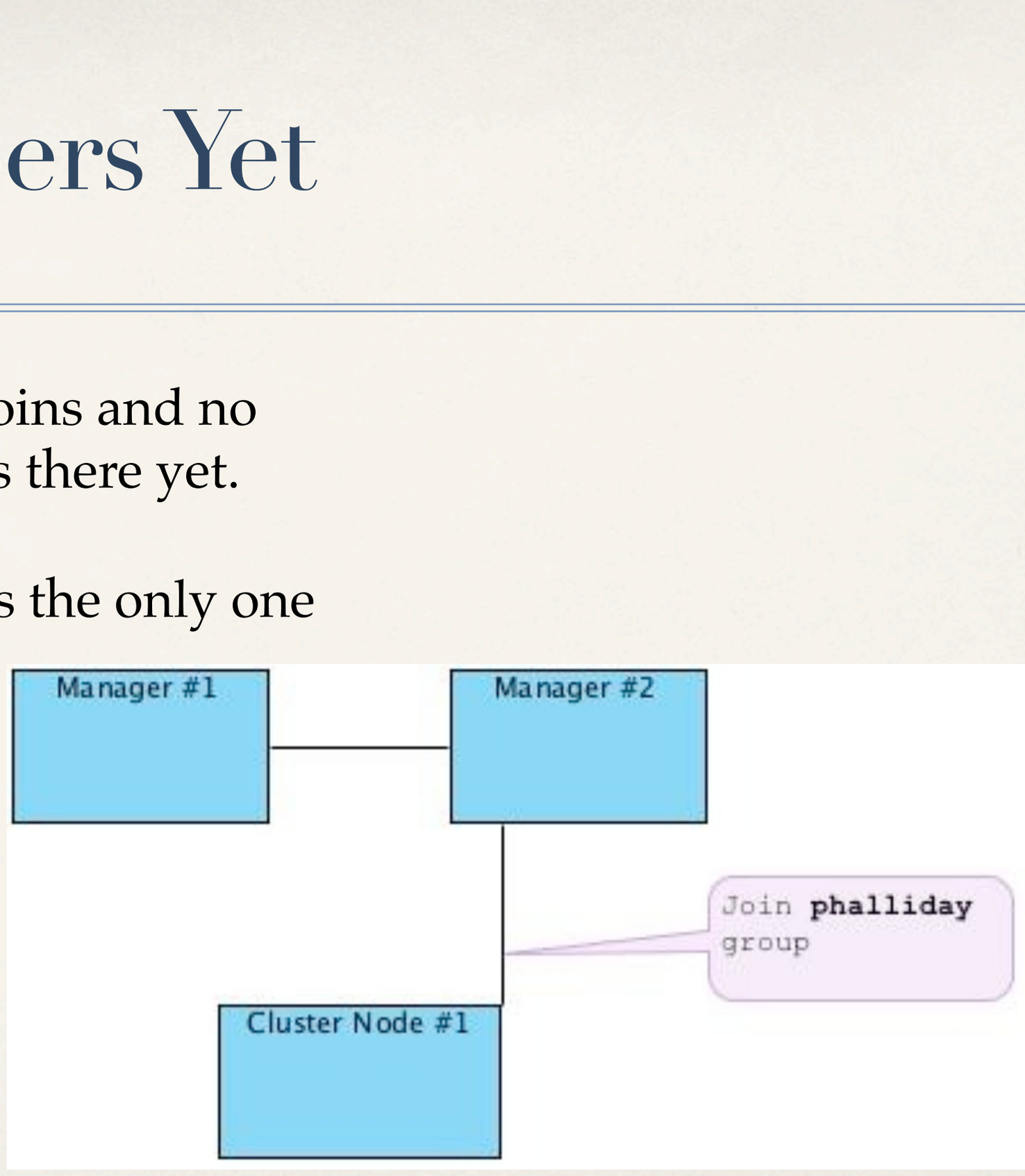
- Startup Management Cluster and ensure there are no groups
- Startup the Messaging Cluster and ensure the new groups get created.



No Members Yet

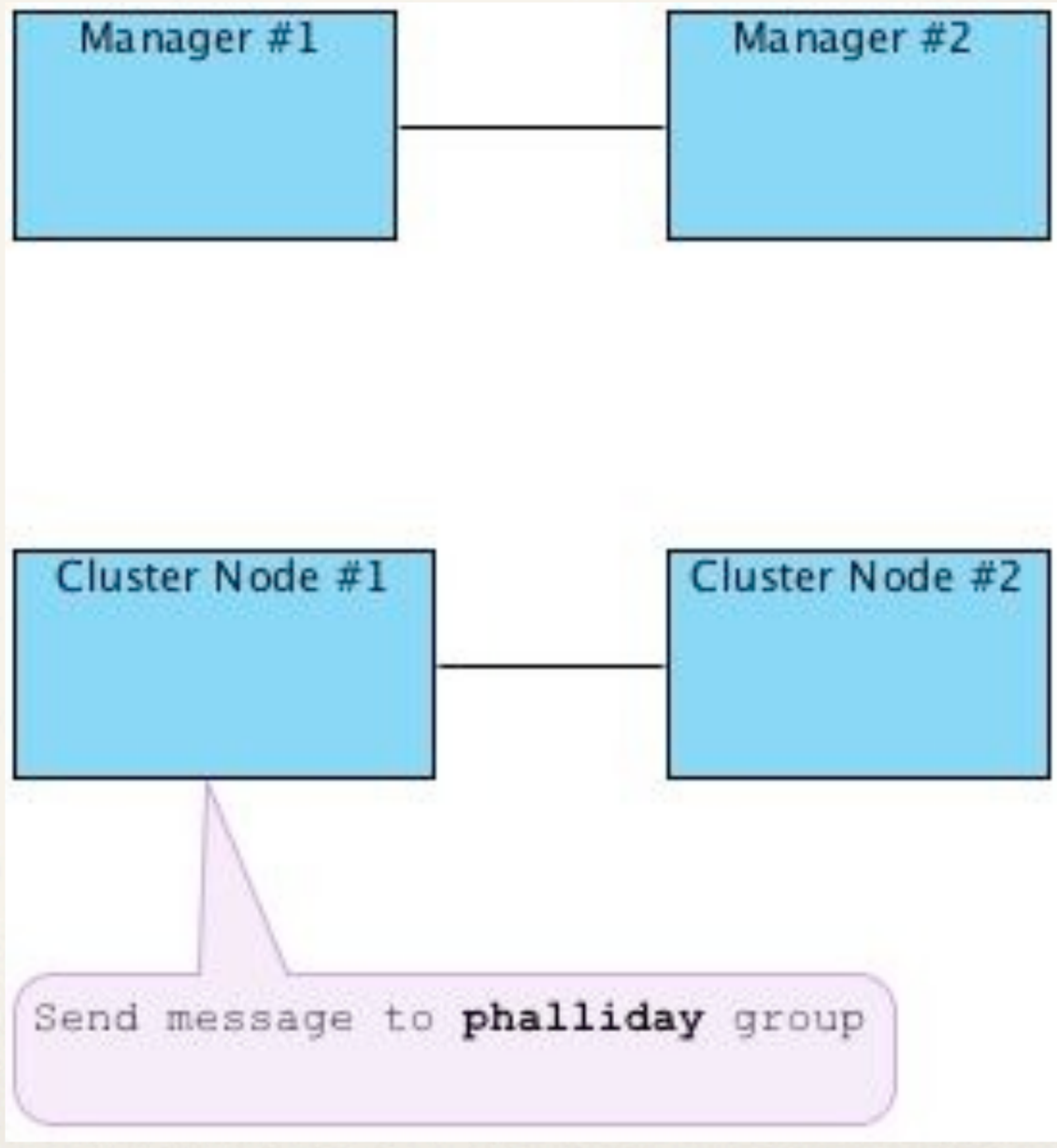
- Cluster Node #1 joins and no one in the group is there yet.

- Verify that Node is the only one in the list



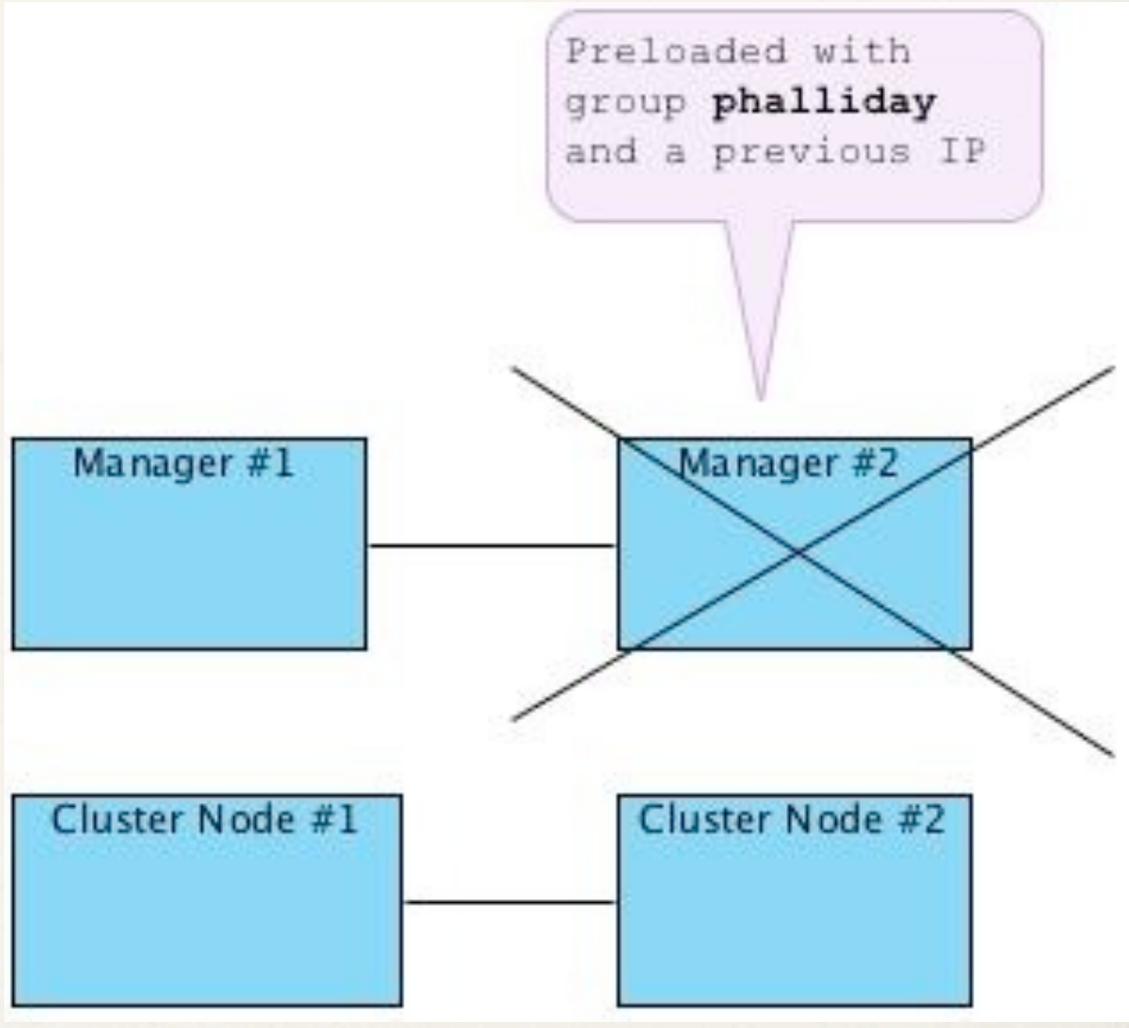
Multiple Group Members

- Cluster Node #1 received a gossip message
- Verify that Cluster Node #2 sees that gossip message.



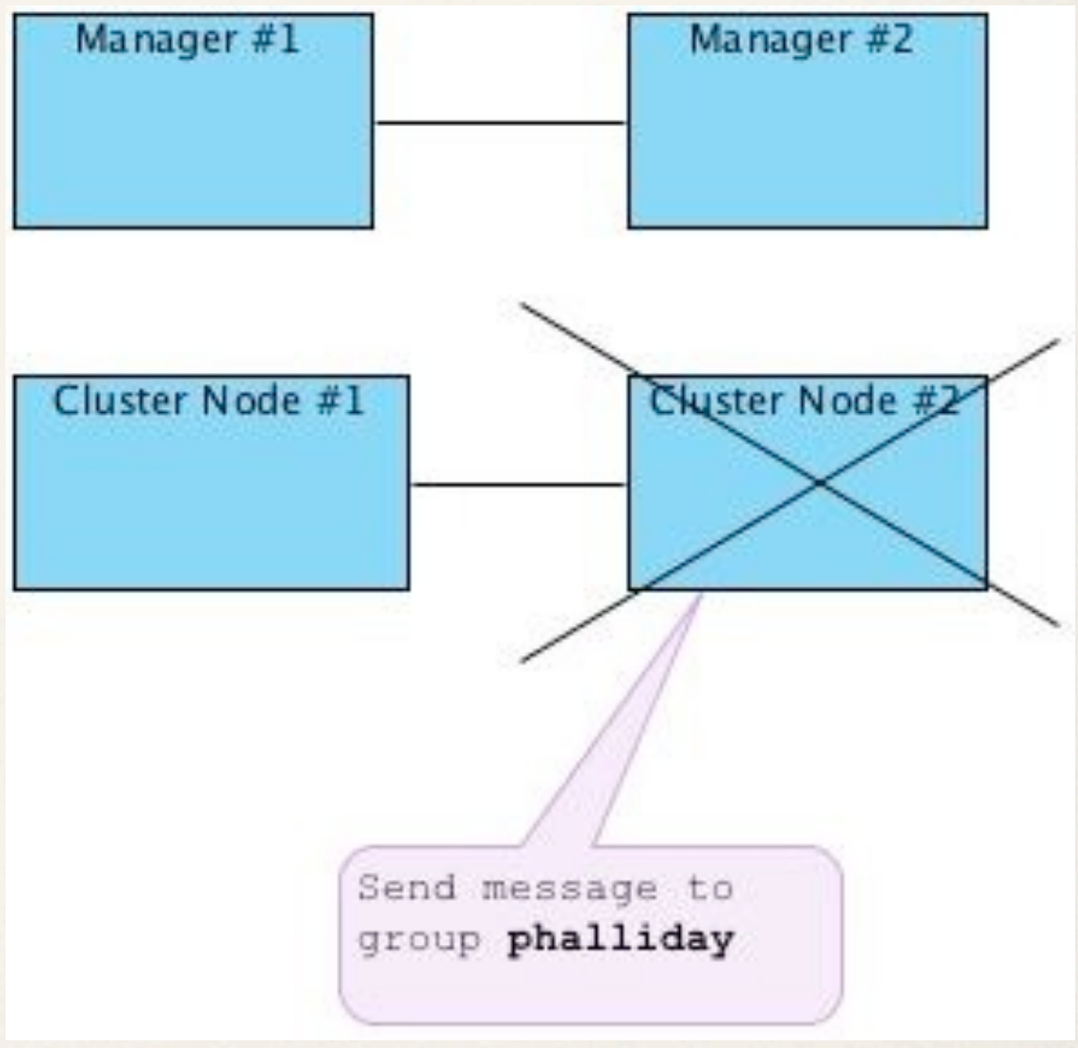
DHT Member Dies

- Manager Node #2 loaded with previous group data, then dies.
- Verify that it can still be accessed on Manager Node #1



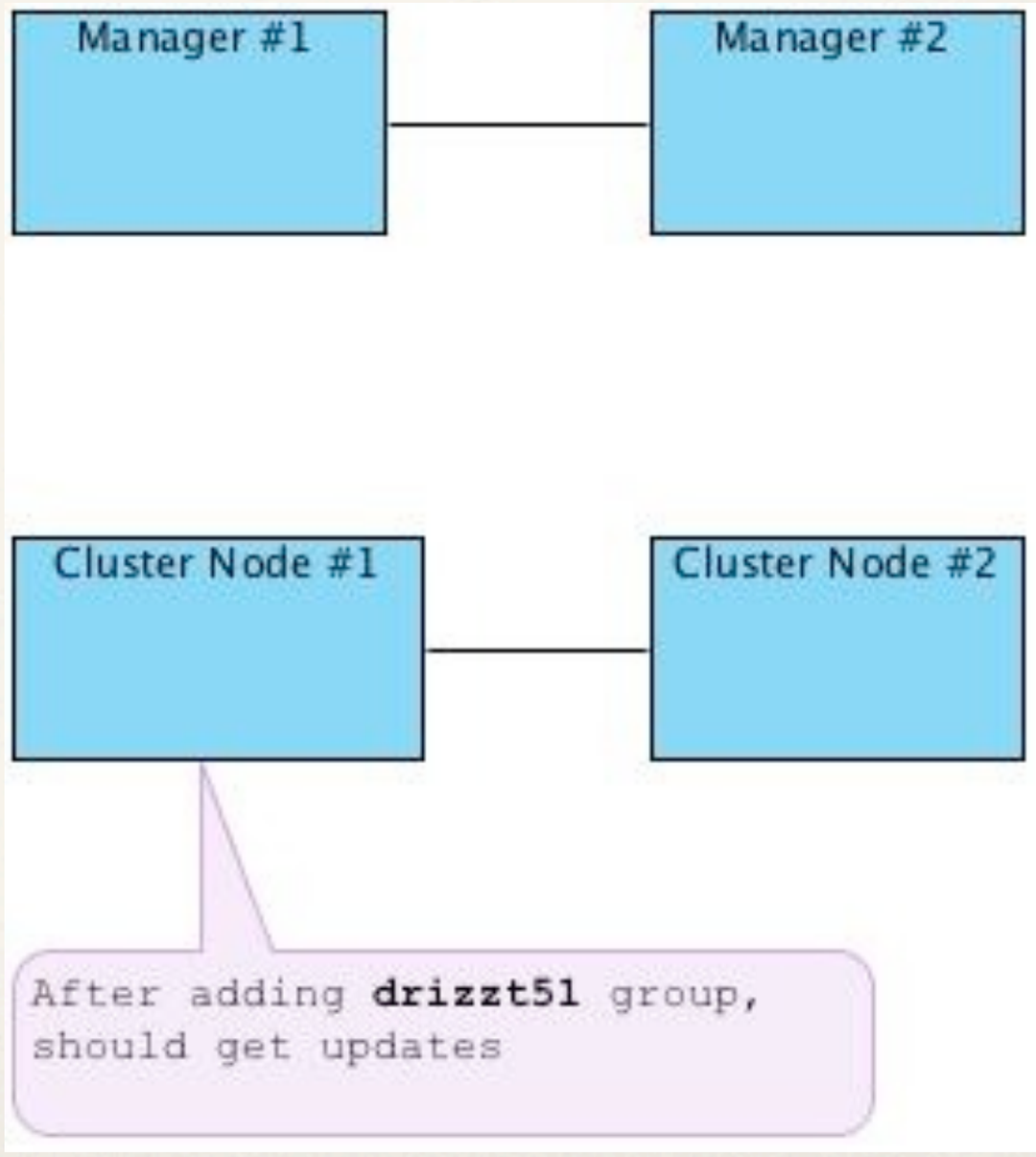
A Gossiper Dies

- Message is sent to Cluster Node #2, and then it dies
- Verify it is still seen on Cluster Node #1



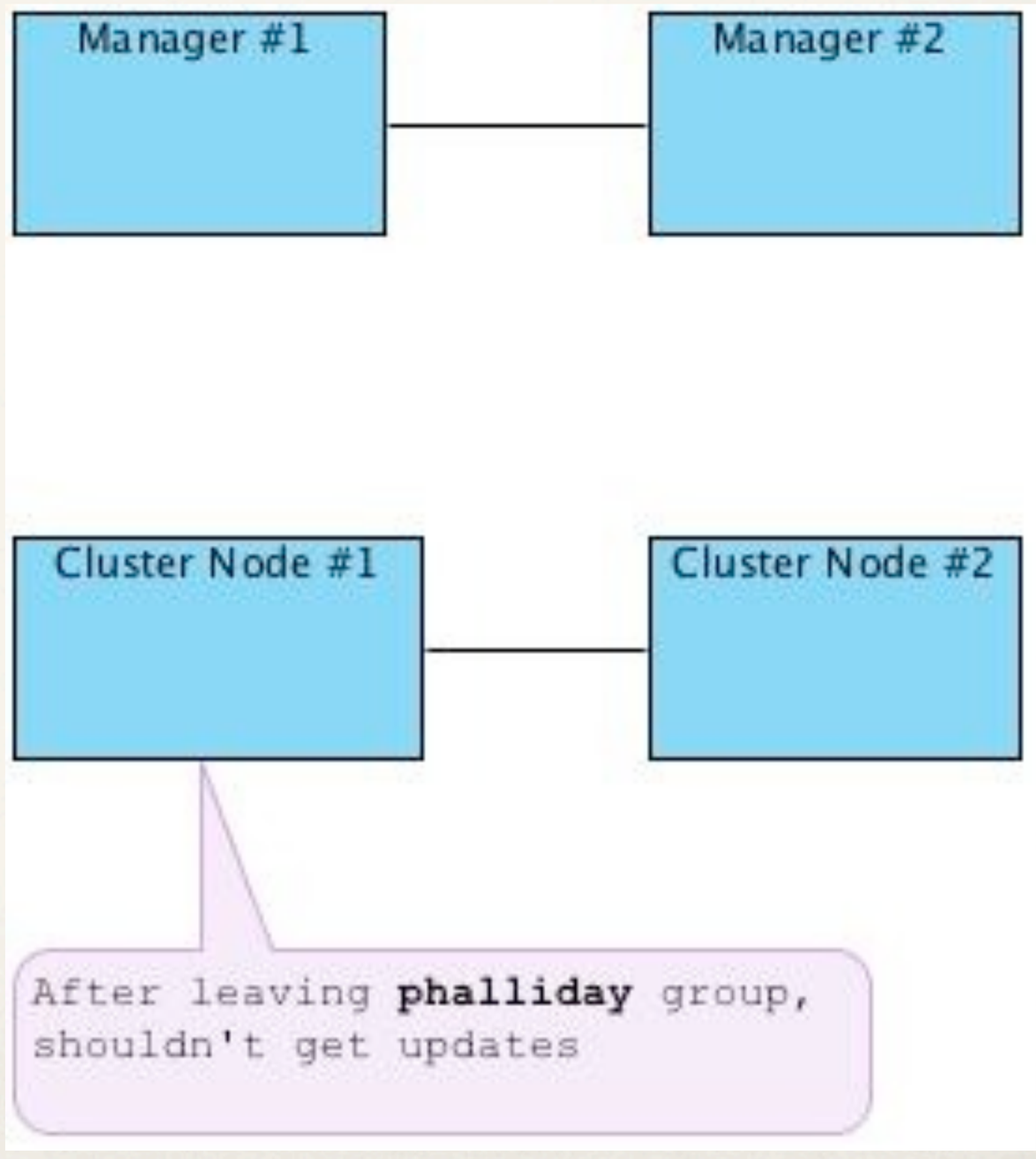
Member Adds New Group

- Cluster Node #1 starts off not in drizzt51 group.
- Verify that the node can't received messages from the group.
- Cluster Node #1 adds new drizzt51 group.
- Verify that they can now get the message of that group.



Member Removes Group

- Cluster Node #1 starts off in phalliday group.
- Verify it can get that group's messages.
- Remove Cluster Node #1 from phalliday group.
- Verify they can no longer receive that group's messages.



Project Documents

- https://github.com/hoangelos/Herd
- https://github.com/hoangelos/gevent-dht
- https://github.com/hoangelos/elastica

Questions / Answers