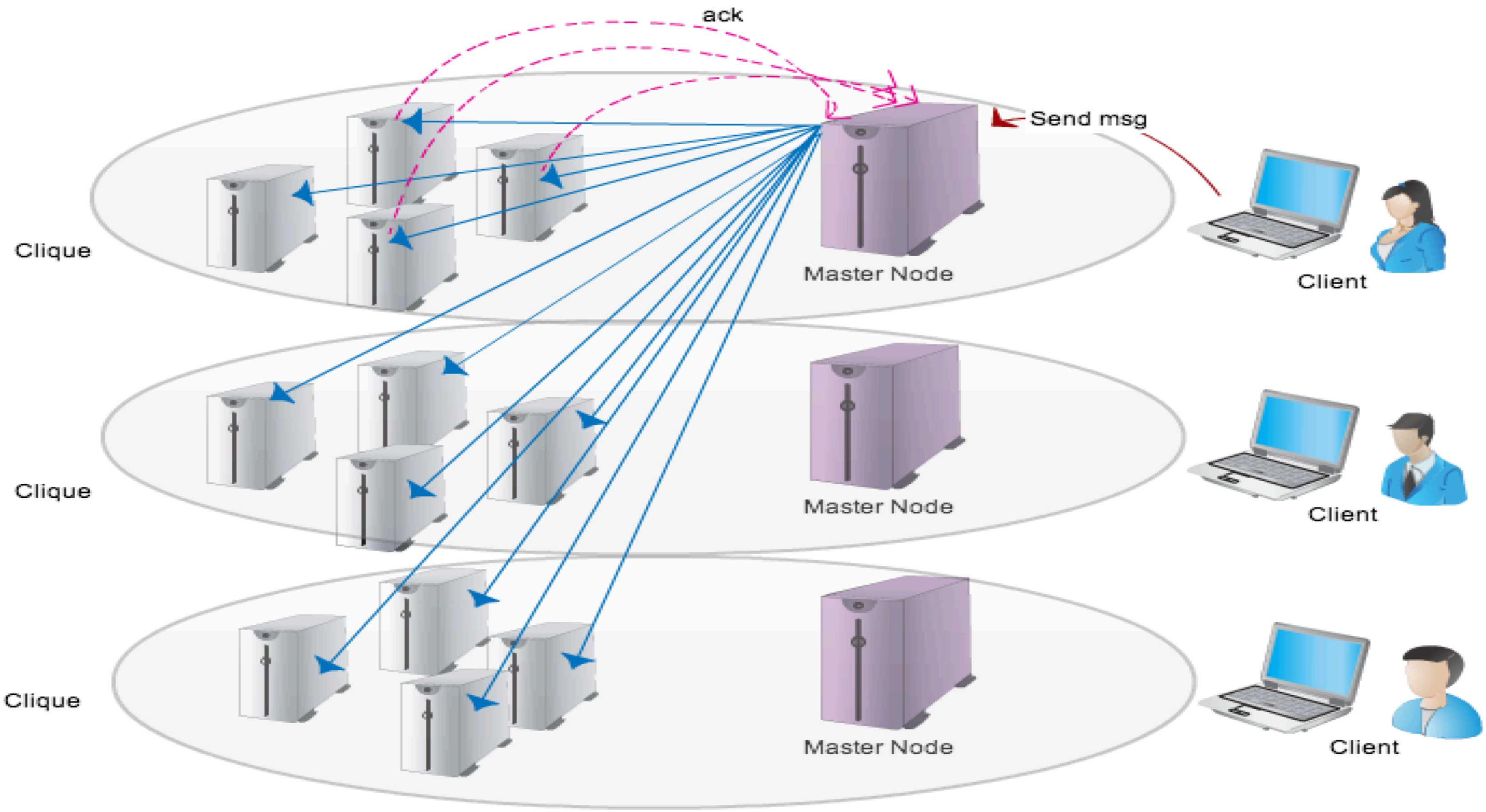
Cse 223b – Spring13

ALI ASGHARI , AMELL ALGHAMDI

System in brief

- Use multicast to send messages between nodes.
- Every host belongs to an ACKing clique where when a node receives a message from another node in the clique, it must ACK that message
  - Other nodes do not reply
  - ACKs are not multicast; point to point
- If node detects missing messages uses gossip protocol to retrieve
- Duplicate messages and conflict detection is left for the application to deal with

System ARCHETECTURE



System Design

The Master Node

- The system will consist of number nodes each is independently accessible to outside clients.

- When a client accesses a node, that node is known as the Master Node for that client.

- At any given time there will be at most as many Master Nodes as clients in system, but may be fewer as multiple clients may share the same Master Node.

System Design

The Clique Nodes

- When a Master Node multicast a message to all nodes in the system, clique mates must reply

- If a clique member fails to reply Master Node re-multicasts resetting to the timeout randomly

- In the event that n clique-mates cannot be found, the Master Node, using a callback, will report the issue to the application

System Design

The Clique Nodes

- The nodes of the system will be organized into cliques of size n. A Master Node within a clique will multicast client requests to all nodes in the system
  - Will only require a response from member of its clique.
- If a clique member fails to reply Master Node re-multicasts resetting to the timeout randomly
- In the event that n clique-mates cannot be found, the Master Node, using a callback, will report the issue to the application

System Design

The Non-Clique Nodes

- Nodes outside the clique may or may not receive messages, receive duplicate messages or receive messages out-of-order from Master Node.
- Any messages received can be immediately committed.

System Design

Fault Tolerance

In the event a node detects it is missing data via the message timestamp:

1. Start with clique mates
2. Contact fixed number of nodes randomly
3. Lastly if all else **fails**, contact the original sender
   - If a request makes it all the way to the Master Node, it will be multicast to all.

Goals

- Provide higher probability of consistency amongst clique nodes and a while allowing for a lower probability of consistency between the rest
  - Since using multicast resending to clique ensure nodes around clique member have high chance of hearing too
- Easy to use interface

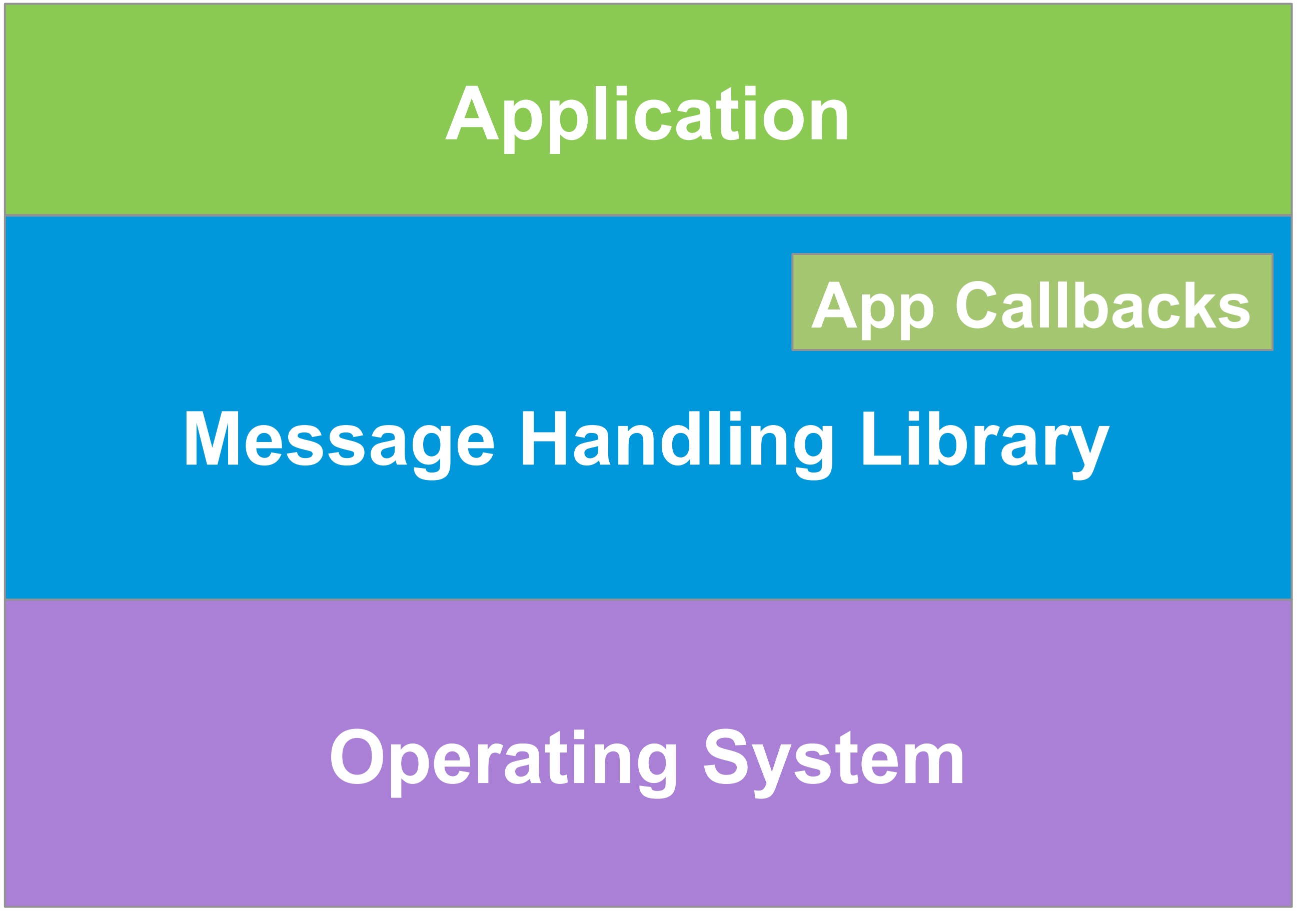
SYSTEM PROPERTIES

- Provide Eventual consistency with high availability
- Guarantees a higher probability of consistency among a small group of clique nodes
- Passes duplicate message handling to application
- Passes conflict resolution to application for handling

Application Usage

- The System is built as a message handling library that the application runs on top of
- Library provides send, handle message, and clique distress API
    - void sendMessage(...): for sending messages
    - HandleMessage: callback into application's when system receives a message
    - Clique Unreachable: callback into application letting it know not every node is clique got message

Application Layers



Application

App Callbacks

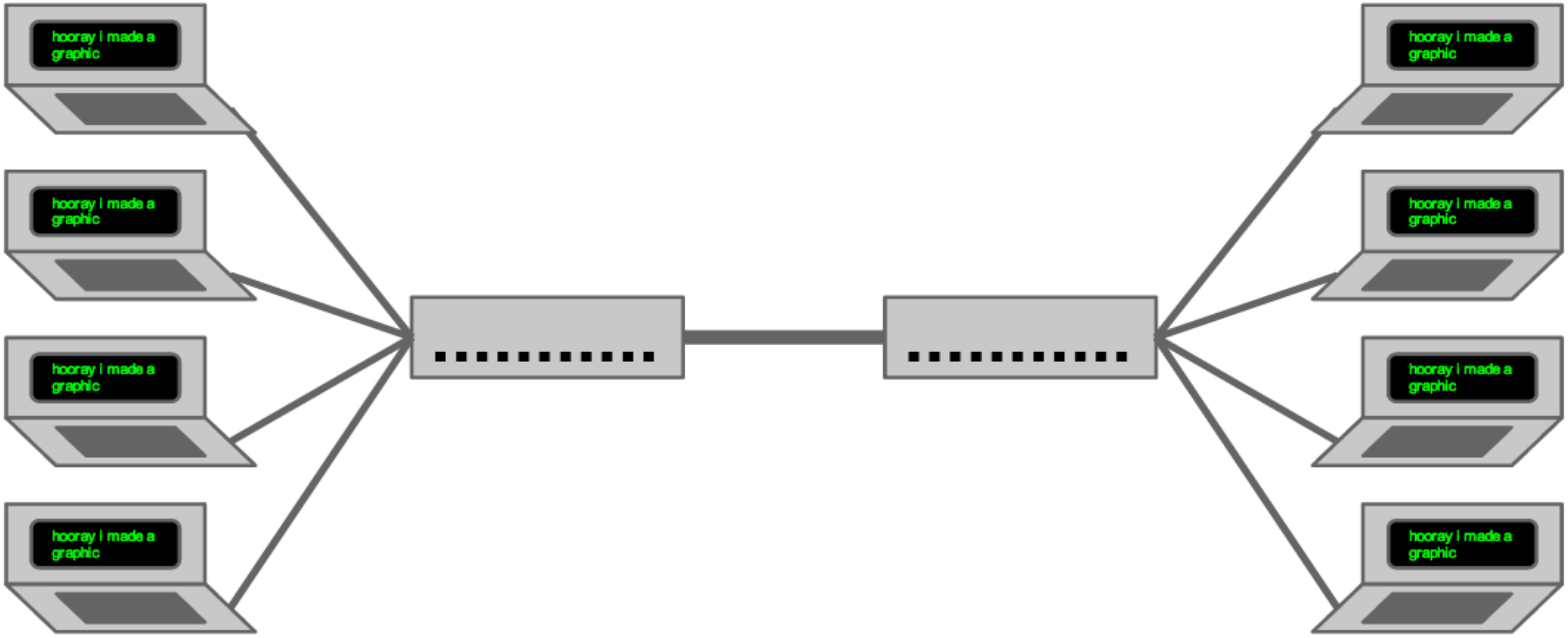
Message Handling Library

Operating System

Sample Application

- Build a Key-Value store to be used by our sample client application
- Sample Application- Petrol Friend
  - (iPhone Gas Buddy rip-off)
- Clique choice geographical
  - Local nodes need higher probability of consistency
  - Distance nodes do not

Sample Network topology mininet



San Diego

New York

Future Work

- Synchronization Protocol: enable a checkpoint enforcing all nodes are consistent
  - Similar to anti-entropy in Bayou
- clique selection properties:
  - How do overlapping cliques affect system
  - Benefits in overall system consistency from spread clique members.