# CS-438
# Decentralized Systems Engineering

Fall 2021

Week 6

# Decentralized search / routing

- Unstructured: flooding, expanding-ring $O(n)$ per search, $O(n^2)$ overall
  Bubble Storm = $O(\sqrt{n})$ per search, $O(n\sqrt{n})$ total

- Structured: build up state about the network
  efficiency: $O(\log n)$ per search, $O(n \log n)$ overall

  - Search: given name/ID, find address/location - piece of information
    assumes lower-layer that "gets us there"
  - Routing: given name, ID, address/location
    use to communicate w/ target
    find a path to target

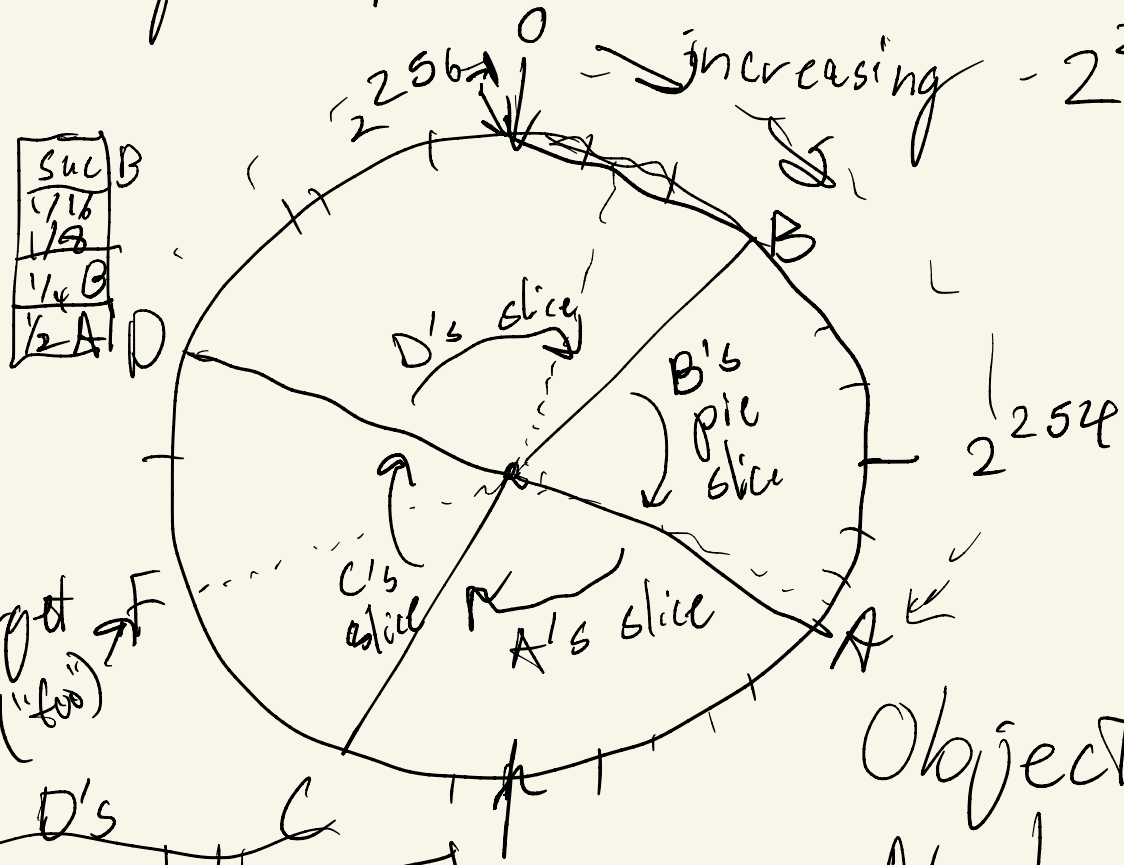
Example search / routing:

- DHTs (search):
  - Chord, Pastry, Kademlia (IPFS, Swarm, BitTorrent, I2P)

- Compact routing:
  - Name-independent: search + routing
    - stretch 1
    - non-compact: $O(n)$ cost per node, $O(n^2)$ overall
    - takes arbitrary names
    - $O(\sqrt{n})$ per node, stretch $\leq 3$
  - Name-dependent: routing only
    - algorithm assigns addresses
    - $O(\log n)$ state per node
    - stretch $\leq O(\log n)$

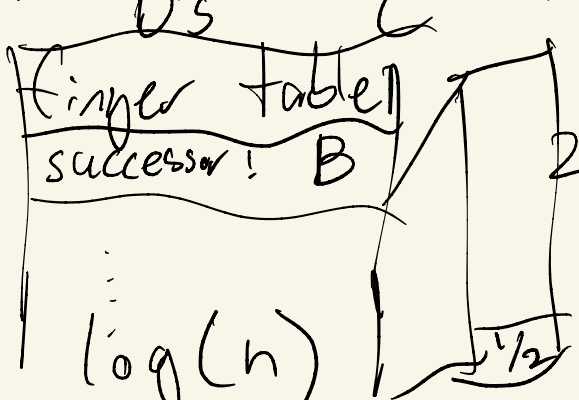
# DHT - Chord O(log n) ⇐

- Hash ID space: HASH(name) ⇒ $H_{ID}$
  e.g. SHA256 ID = 256-bit integer
  $2^{256}$ addrs




Uses hash IDs for: - node IDs
- obj IDs

Object ID: HASH(contents)
Node ID: HASH(pub key)
"self-certifying ID"




(challenge