Distributed File System

Chen Jin

Outline

- Motivation
- System overview
- System implementation
- Ongoing work

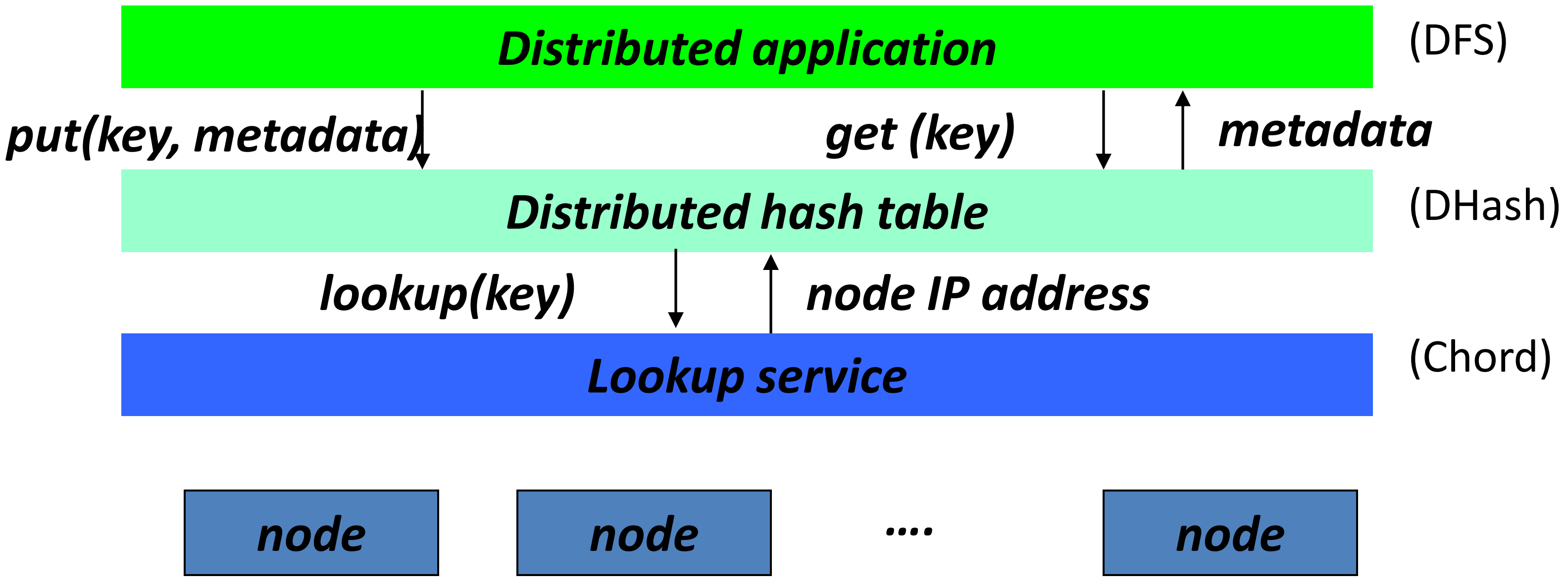
Motivation

- Network storage have received great attention
- A single MDS in Current distributed/parallel FS
- Decentralized metadata management is necessary

System Overview

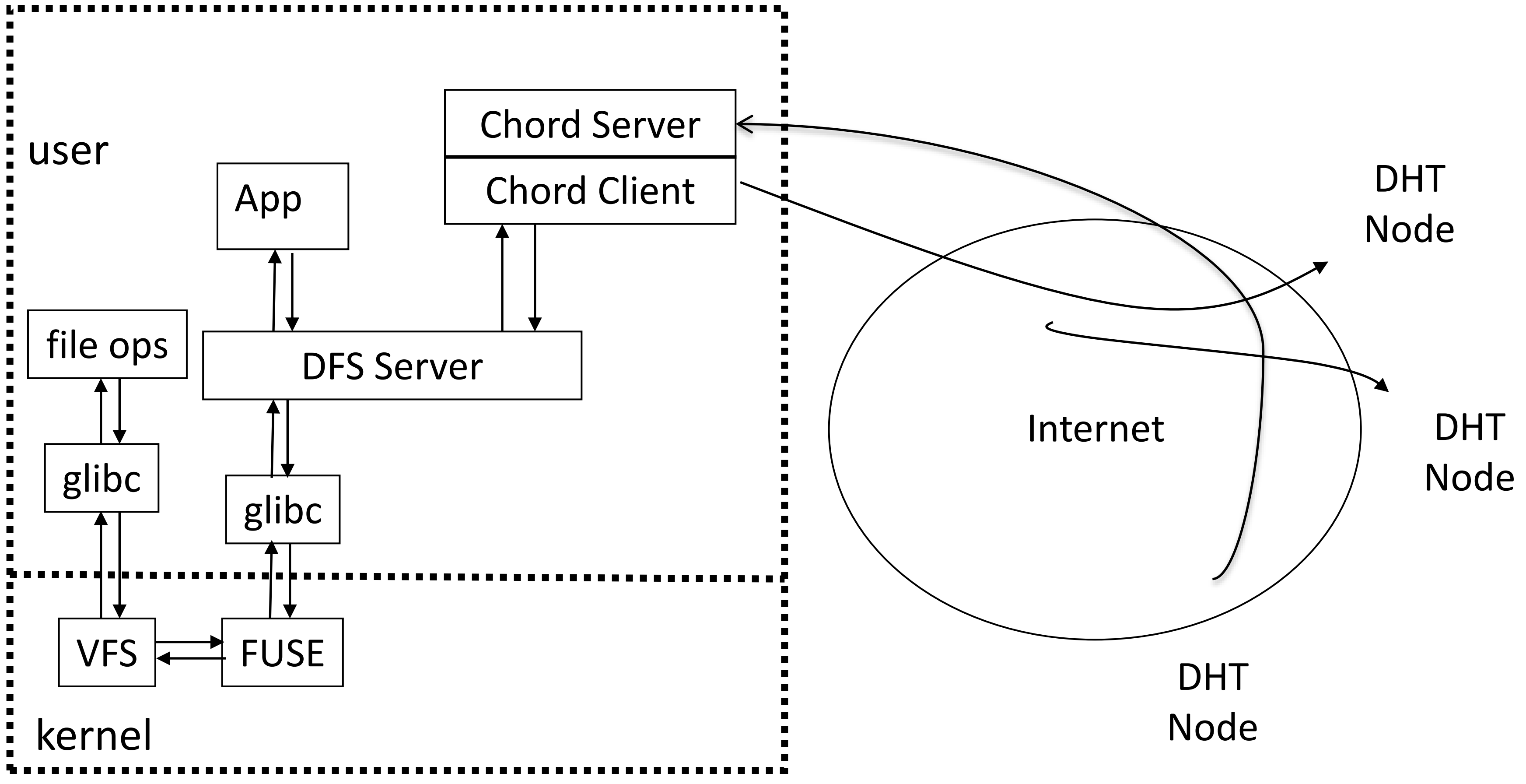
- DHT-based metadata server cluster
  - Chord, Chimera, CAN, Pastry
- User-space local file system
  - FUSE

Software Stack



- DHT distributes metadata storage over many nodes

System Architecture



Lookup service

- Centralized
  - Napster (centralized Database, O(N))
- Flooded queries
  - Gnutella (worse case O(N))
- Routed queries
  - Chord (O(logN))

Chord

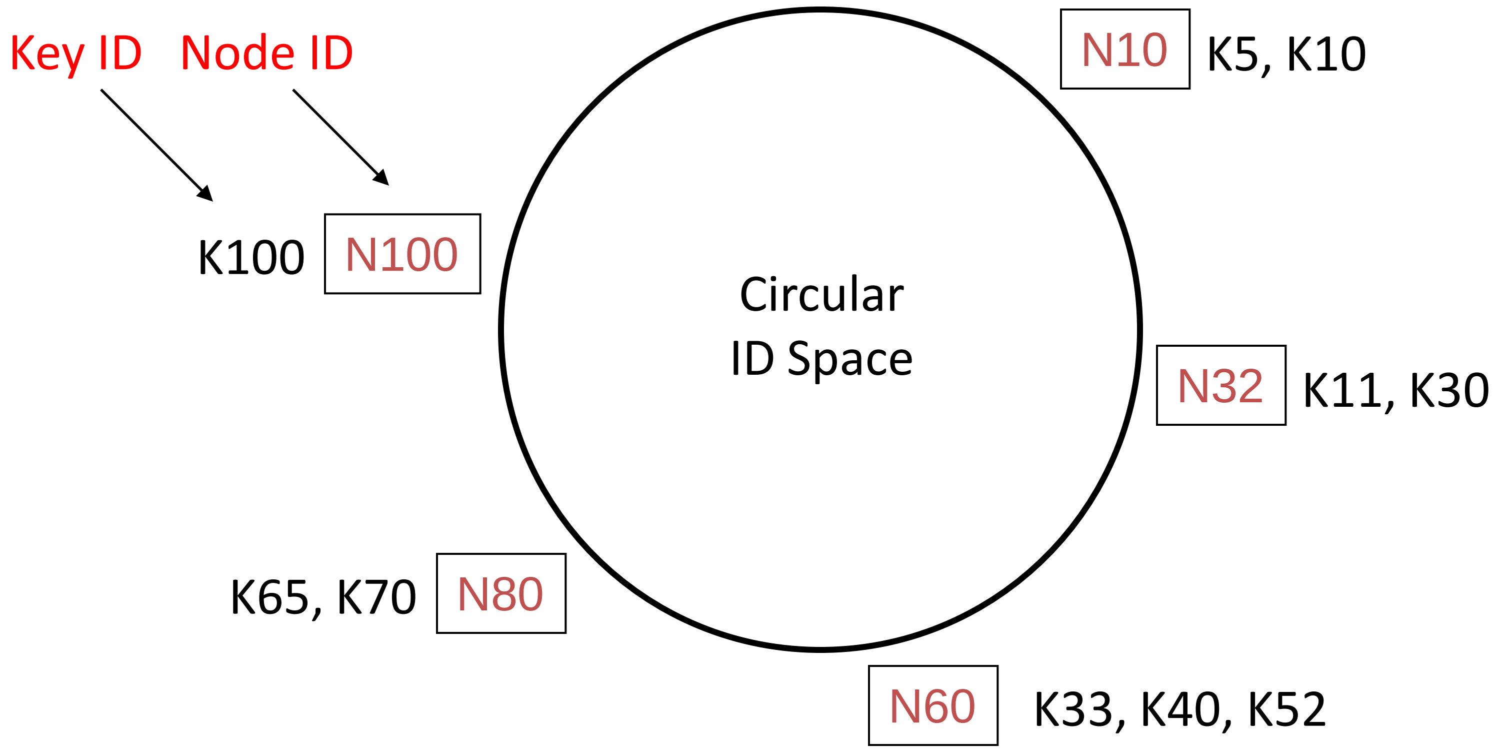
- Chord Implementation
  - Distributed routing table
  - Transport Layer:

    implemented on top of the SFSlite asynchronous RPC libraries over UDP

Chord Cont.

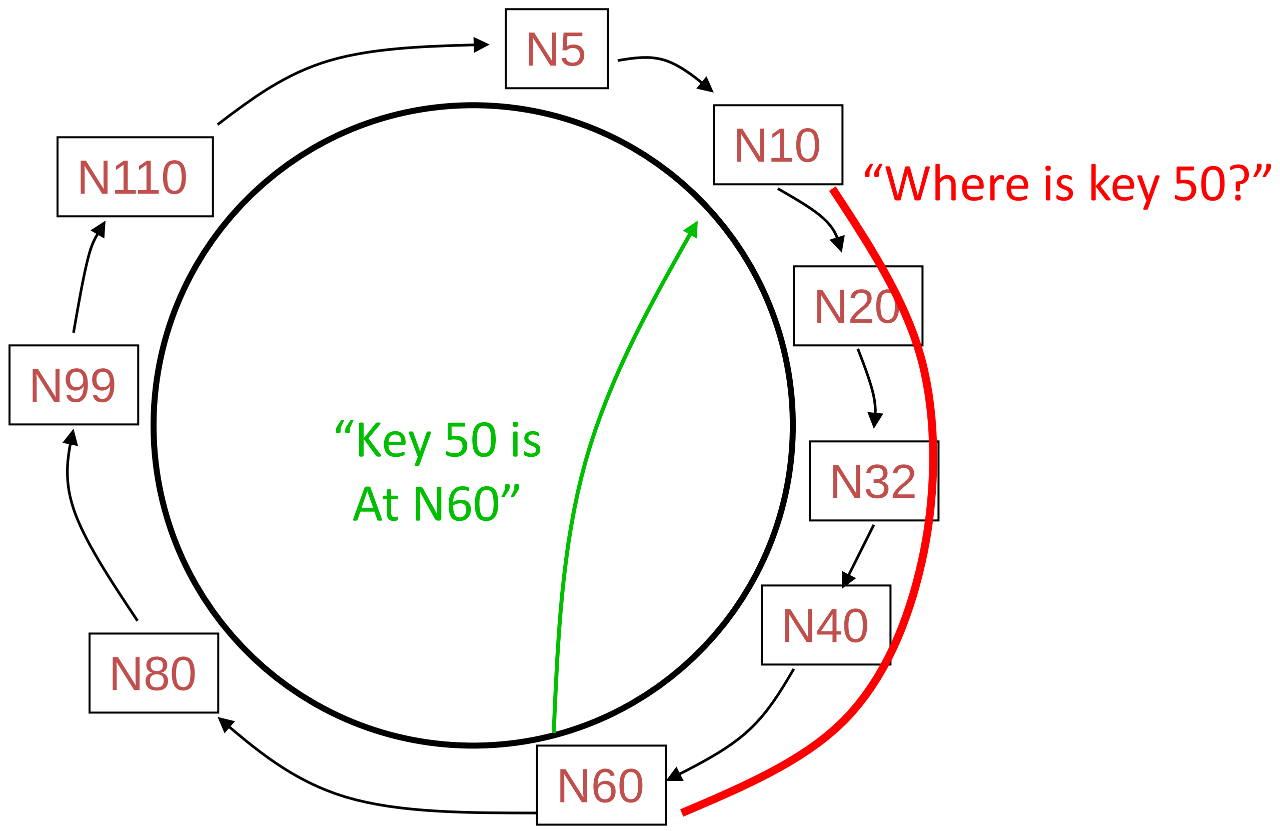
- Chord IDs
  - Chord ID Key identifier = SHA-1(key)
  - Node and filesare assigned key in the same ID space
- Node IDs Arranged in a circle with 2^n-1(n=160)
- Consistent hash
  - filename and IP address can be uniformly distributed in the ID space
  - Nodes join and leave the network without disrupting the network
- How to map files IDs to node IDs?

Chord Hashes a Key to its *Successor*



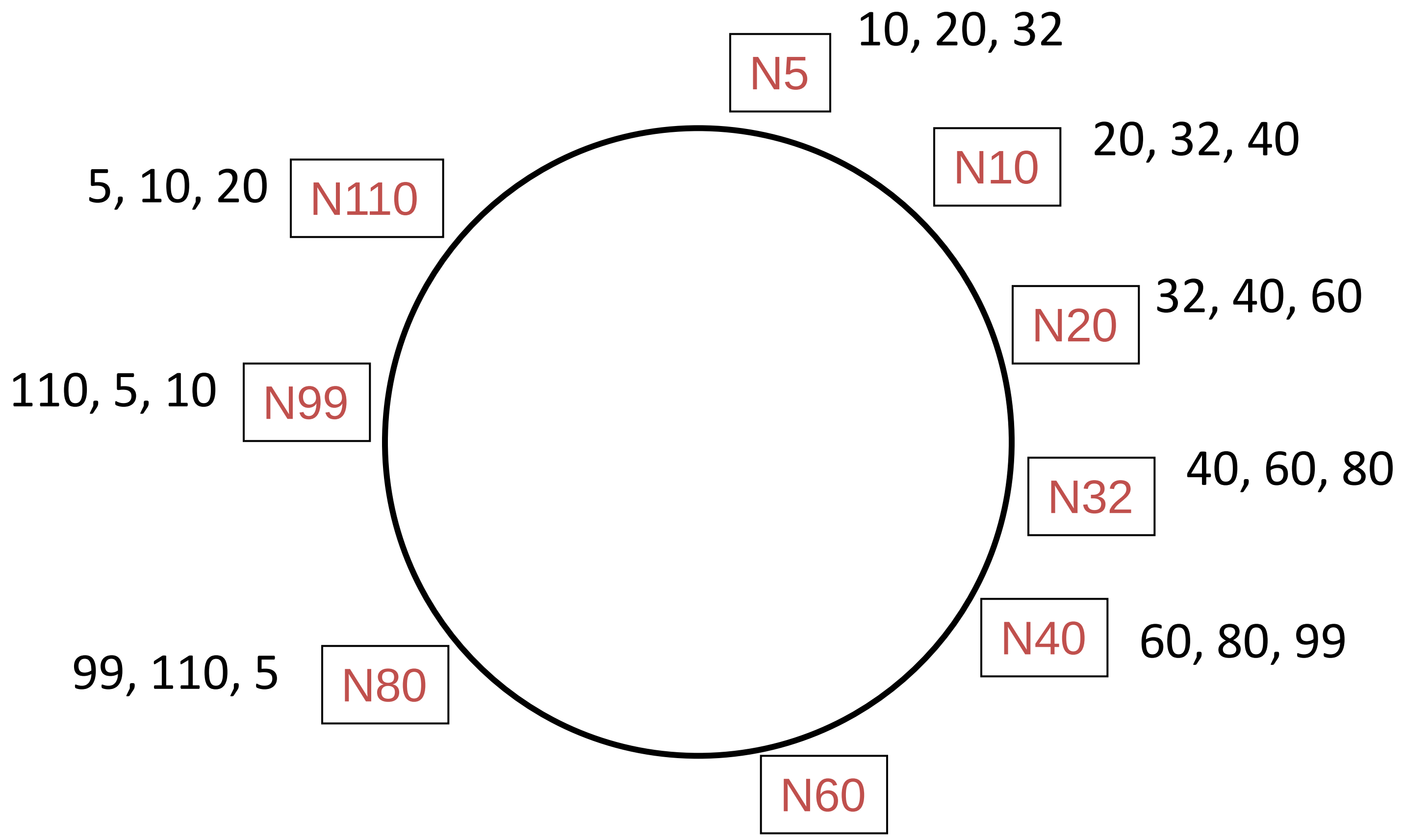
- Successor: node with next highest ID

Basic Lookup



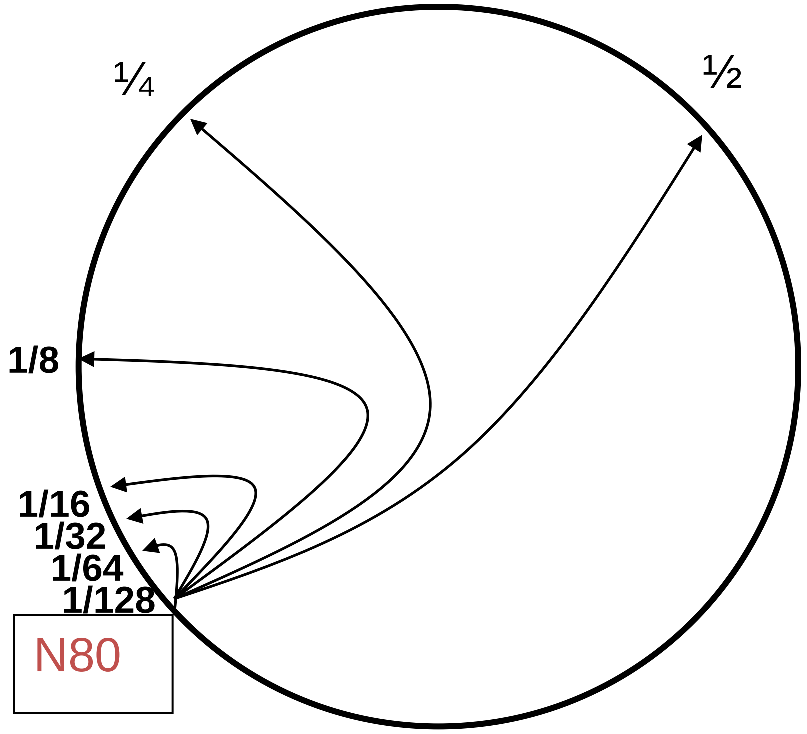
- Lookups find the ID’s predecessor
- Correct if successors are correct

Successor Lists Ensure Robust Lookup

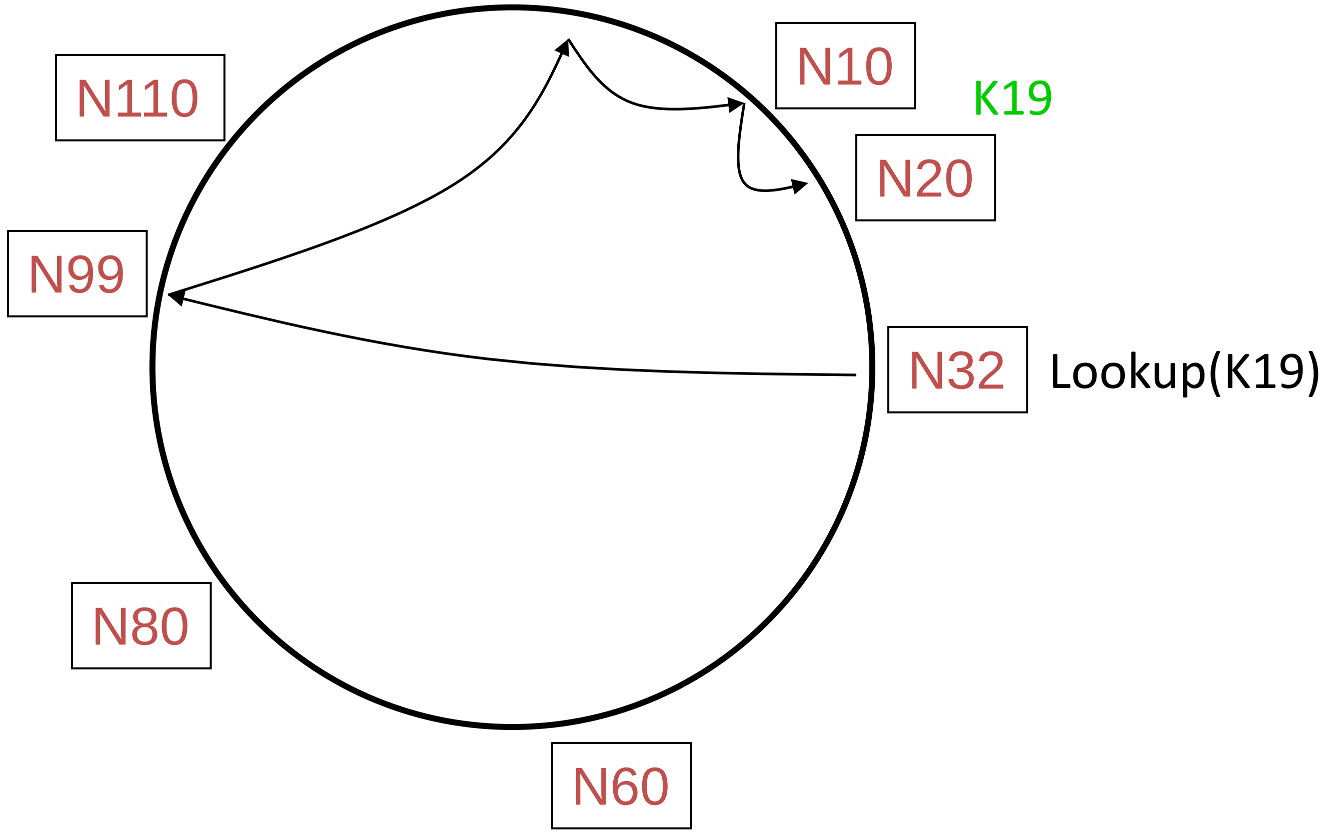


- Each node remembers *r* successors
- Lookup can skip over dead nodes to find blocks

Chord "Finger Table" Accelerates Lookups



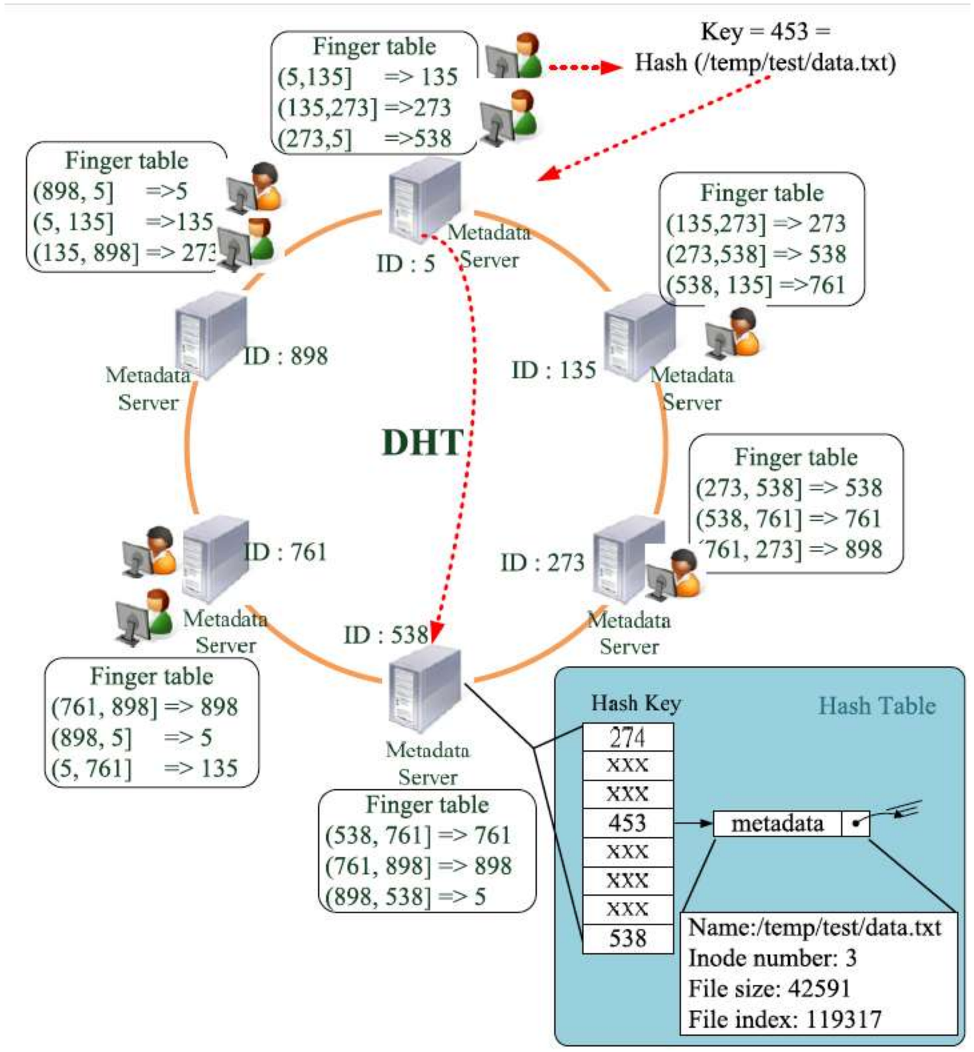
Chord lookups take O(log N) hops



Chord lookup algorithm properties

- Interface: lookup(key) $\rightarrow$ IP address
- Efficient: O(log N) messages per lookup
  - N is the total number of servers
- Scalable: O(log N) state per node
- Robust: survives massive failures
- Simple to analyze

Case Study

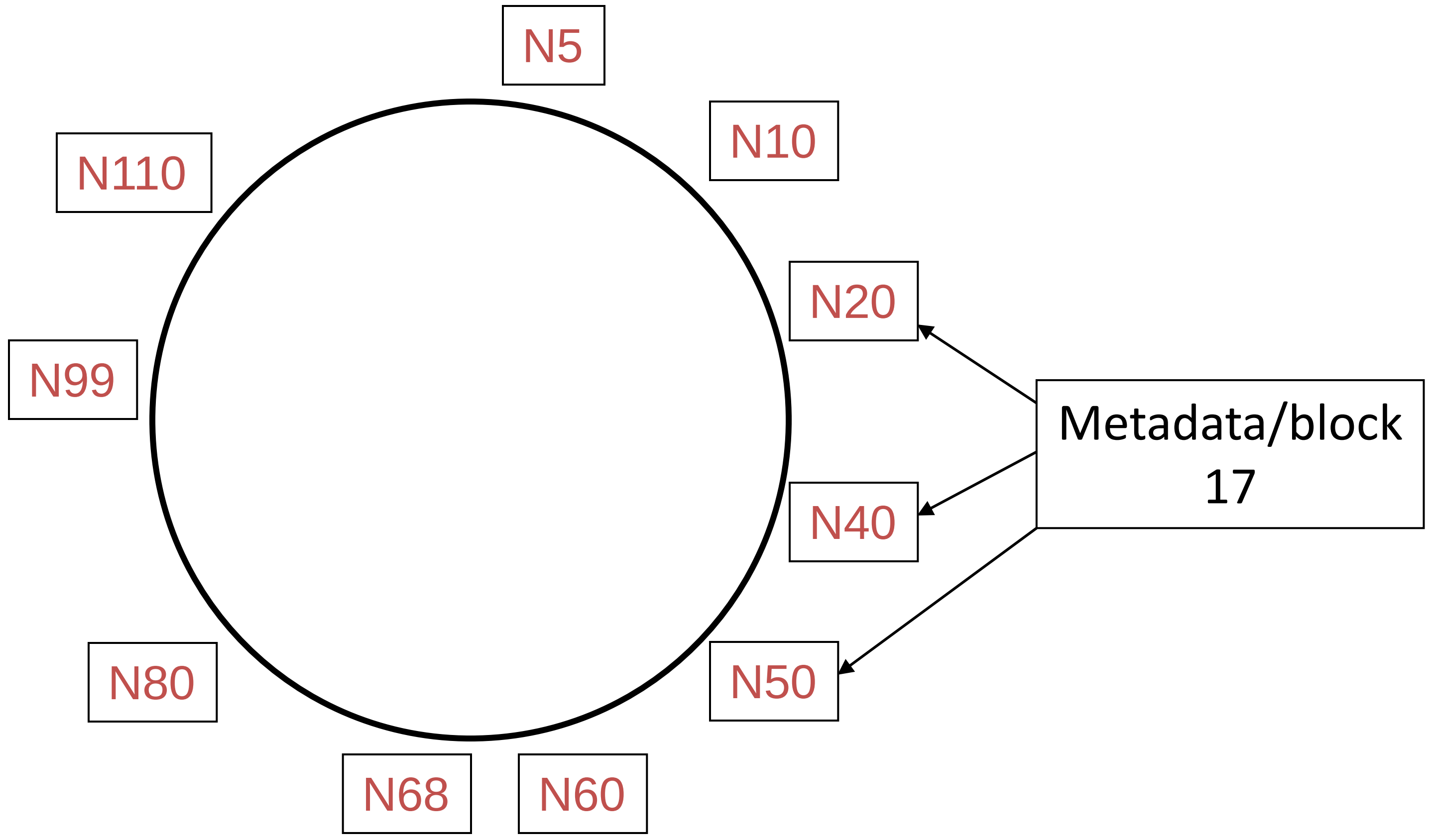


Source: "A reliable DHT-based Metadata server cluster"

Join & Leave the Ring

- Join
  - Sent *Join* message via ID Finger table
  - Until reach the node immediately preceding the joining node in the Chord Ring
- Leave
  - Move the metadata to successors
  - Send out the *leave* message

DHash Replicates metadata/block at *r* successors



- Replicas are easy to find if successor fails
- Hashed node IDs ensure independent failure

Ongoing Work

- Have done
  - Compiled Chord on Falkon
  - Setup a Chord Ring on one node
  - Get/put metadata
- To do
  - Setup a chord ring on multiple nodes
  - Get/put metadata
  - Implement local file system

Performance Evaluation

- Simple LAN Benchmark:
  - Baseline: NFS
  - System setup
    - 8 DHash nodes at Falkon
    - No DHash replication
    - One active writer at Falkon01
    - Whole-file read on open()
    - Whole-file write on close()
  - Performance indices
    - Round-trip times, open/close, read/write, stat