



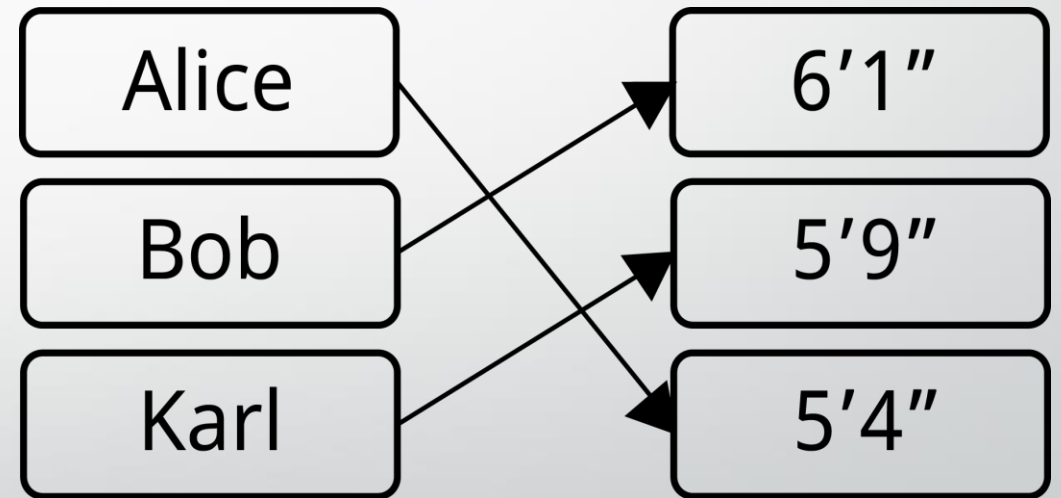
Improving a Hypercube Structured Distributed Hash Table

James R. Wilburn

Mentor: Dr. Udaya Shankar

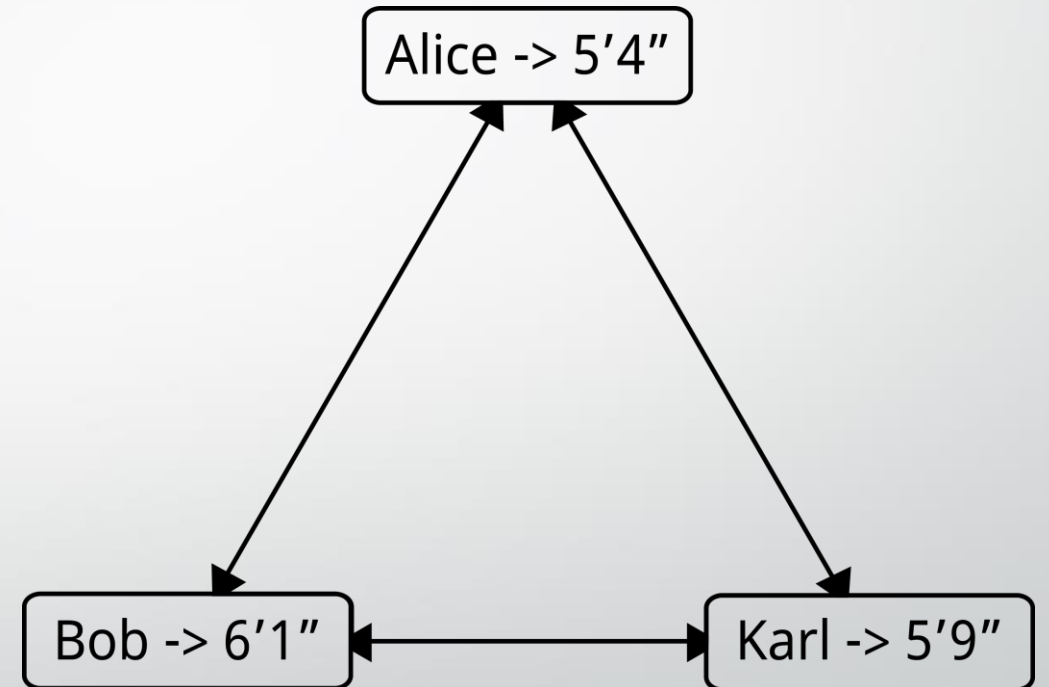
What is a Hash Table?

- Store data
- Map keys to values
- Can *get*, *set*, or *delete*



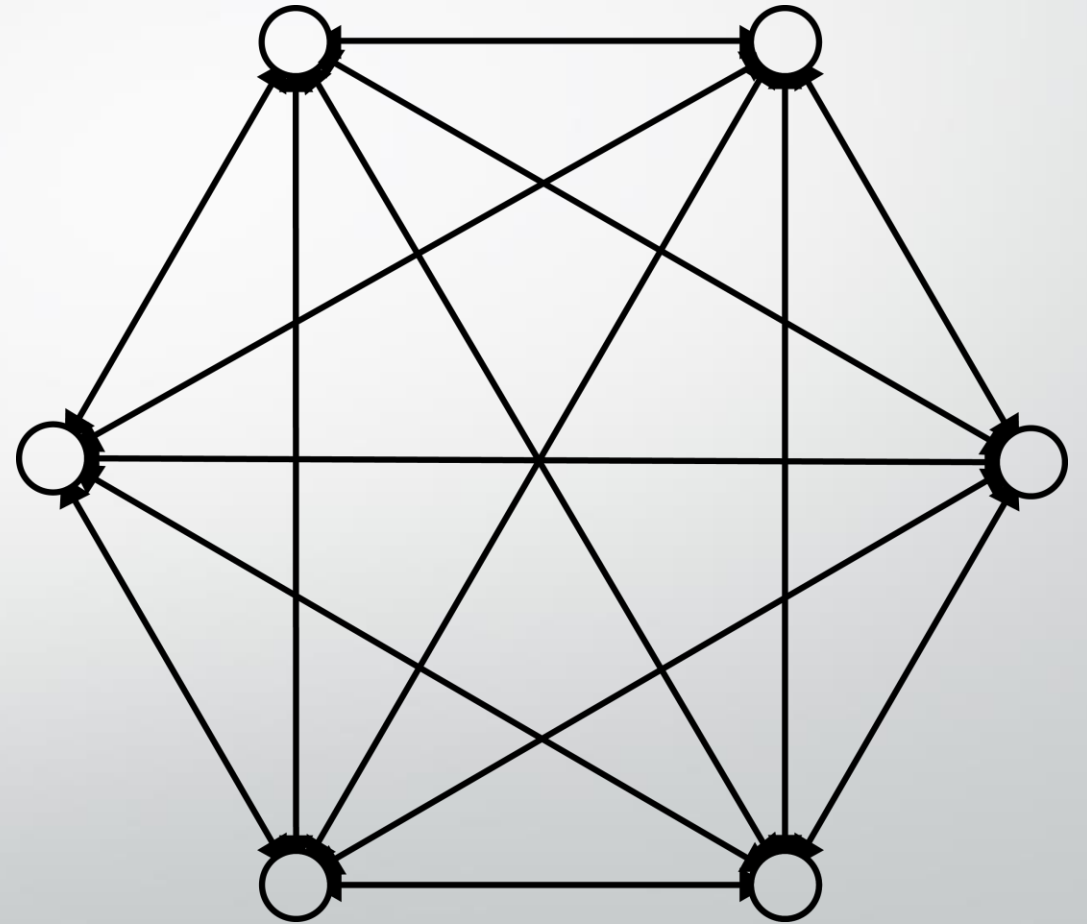
What is a Distributed Hash Table?

- Transparent
- Can still *get*, *set*, or *delete*
- Data spread across a network



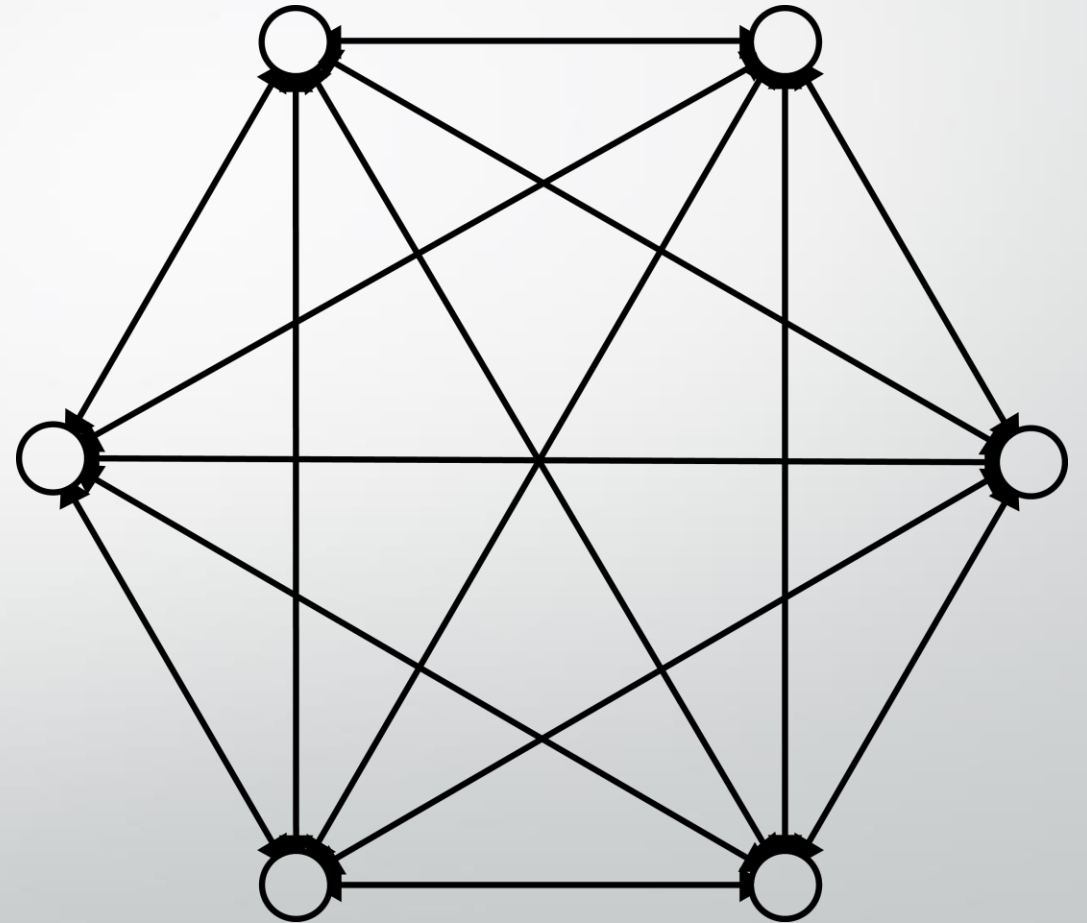
Unstructured Clusters

- n vertices
- $n-1$ edges per vertex



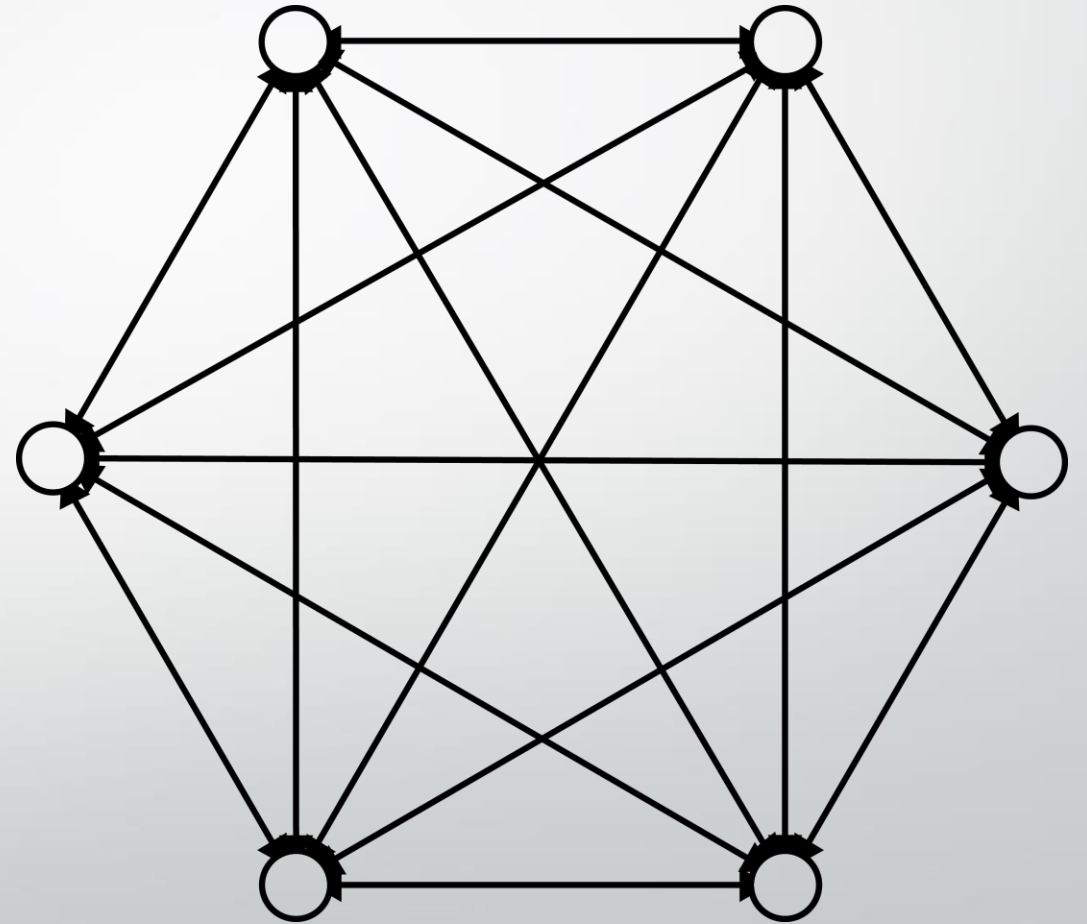
Advantages

- Max distance of 1
- Easy to maintain when small
- Adapts to change
- Redundant links



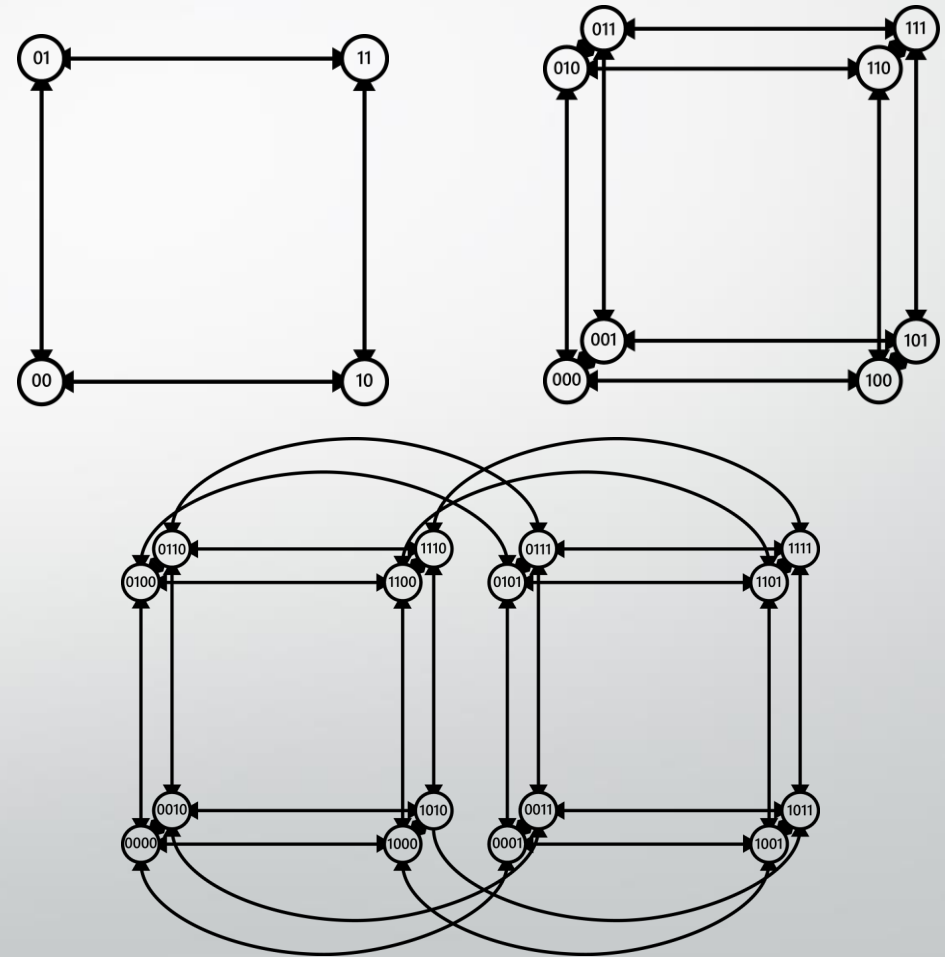
Disadvantages

- Slow at large sizes
- Search is slow



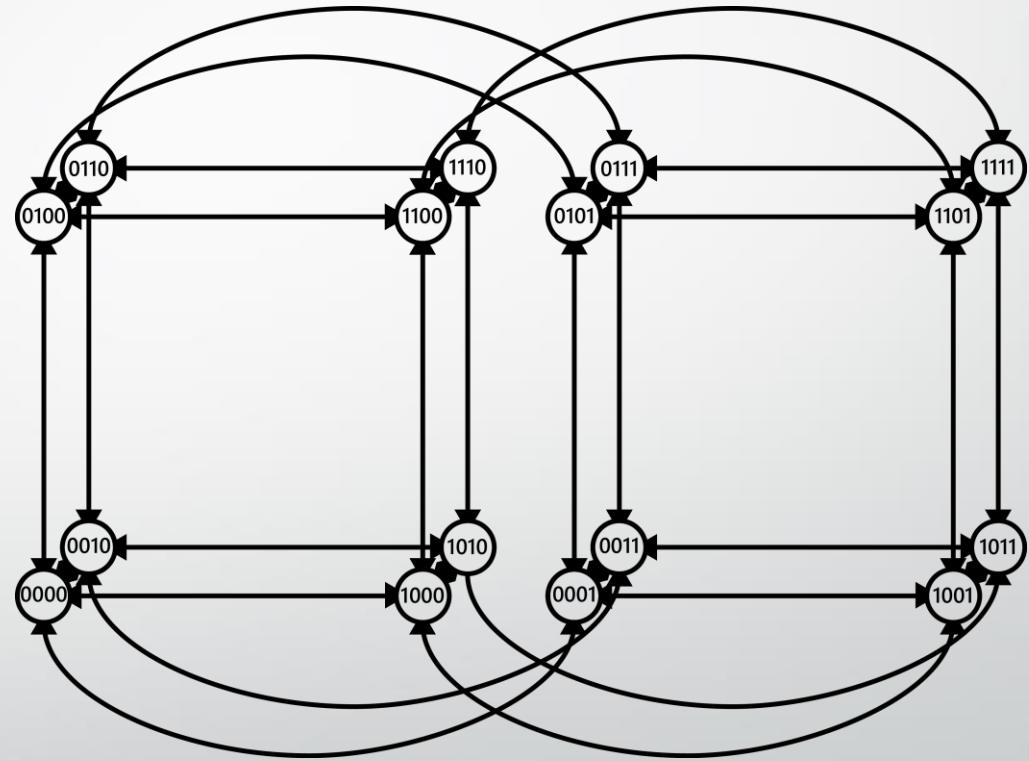
Hypercubic Networks

- 2^n vertices
- n edges per vertex
- Recursive design
- Labelled vertices



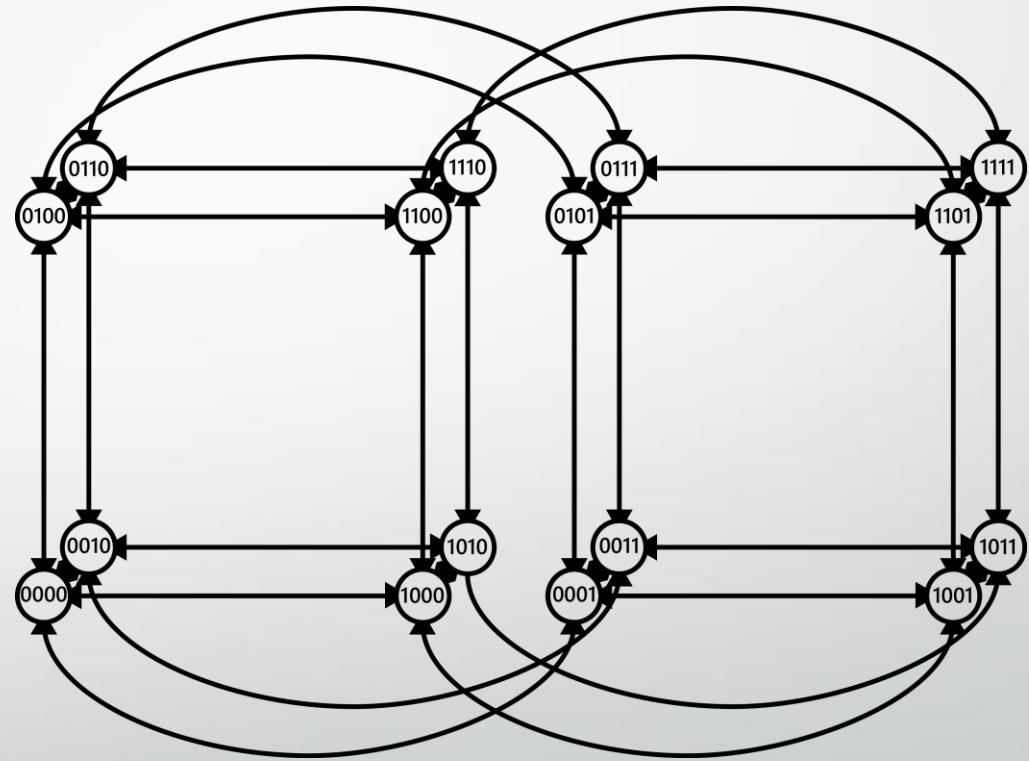
Advantages

- Short max distance
- Redundant links
- Dynamic growth
- Search is fast



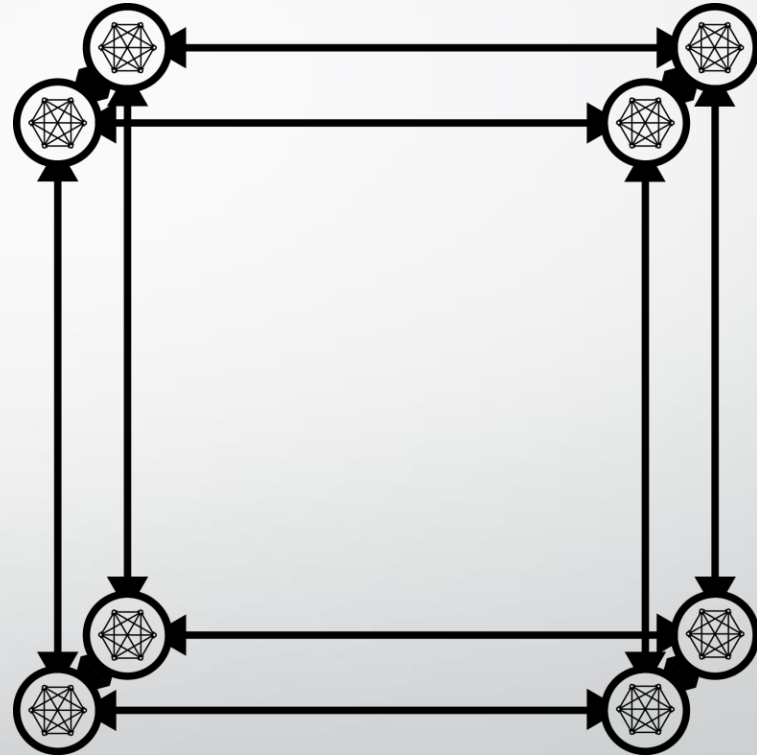
Disadvantages

- Hard to maintain
- Weak to change



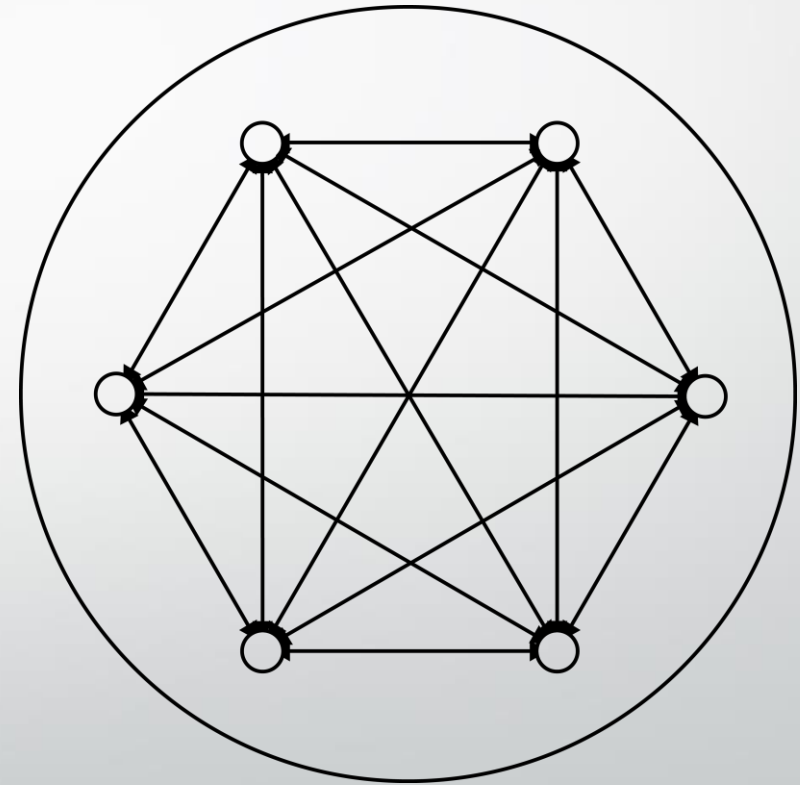
PeerCube

- Cluster at each vertex
- Cluster data replication
- Limited cluster size



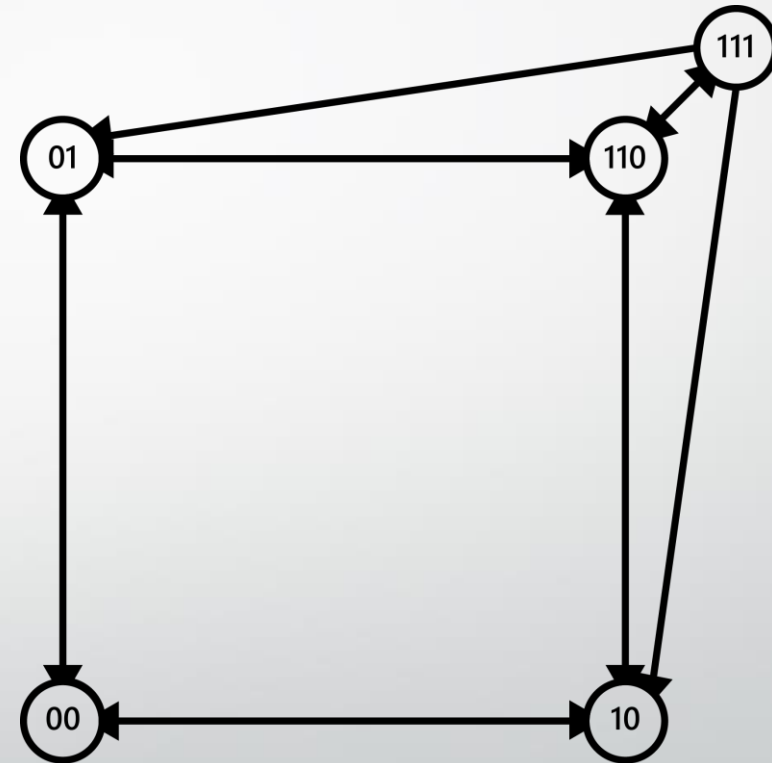
PeerCube Clusters

- Grow and Shrink
- Merge and Split
- Share data internally
- Smin and Smax



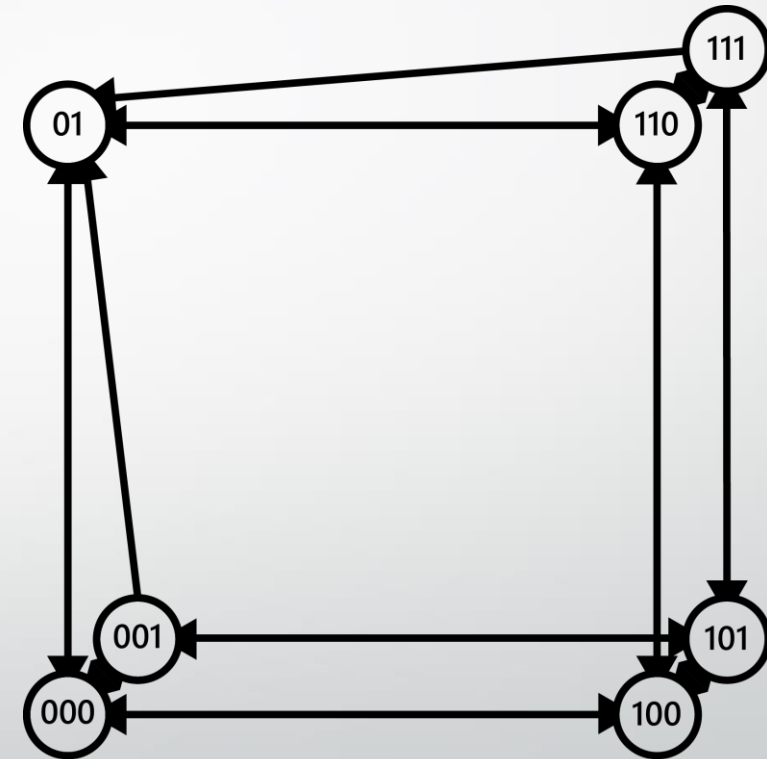
Cluster Splitting

- When the cluster is larger than S_{max}
- Makes two clusters smaller than S_{max}
- Links can be unidirectional



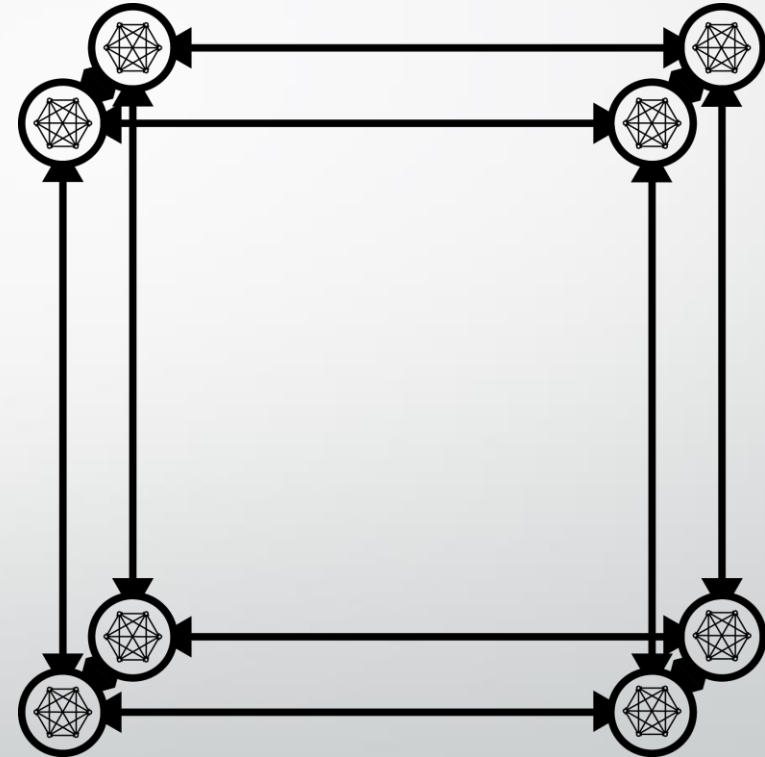
Cluster Merging

- When the cluster is smaller than S_{min}
- Makes one cluster larger than S_{min}
- More than two clusters can merge



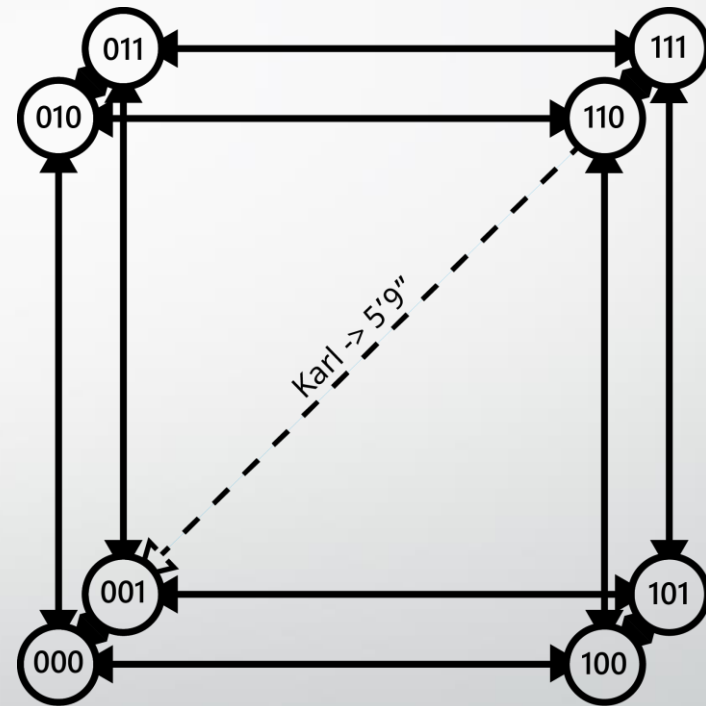
Problems

- No data prioritization
- All data is in memory



Hot Data Caching

- Cache frequently accessed data
- Preempt structure
- Key-specific links
- Latency decrease



Cold Data Paging

- Page infrequently accessed data to disk
- Saves RAM
- Minimal latency increase

