Processing of Query in Peer to Peer Networks

R.Saravanan

M.Tech**,** Lecturer

Department of Computer Science Engineering

St.Joseph College of Engineering and Technology

P.O.Box:11007,

Dar Es Salaam, Tanzania.

ABSTRACT

DHT based peer to peer network system is a general range of query scheme. These schemes can support the range of query without modifying the underlying DHTs and they cannot guarantee to return the query results with bounded delay. The query delay in these schemes depends on both the scale of the system and the size of the query space. This paper, proposed Armada, an efficient range of query processing scheme to support the delay-bounded single-attribute and multiple-attribute range queries called Armada. Describe the order-preserving naming algorithms for assigning adjoining ObjectIDs to objects with close attribute values. The design of forwarding tree is to efficiently match the search paths of range queries to the underlying DHT topology. Based on the tree, two query processing algorithms are proposed respectively single-attribute and multiple attribute range queries within a bounded delay. Analytical and Simulation results shows that Armada is an effective general range query scheme on constant-degree DHTs and can return the query results within 2 logN hops in a P2P system with N peers, regardless of the queried range or the size of query space.

Keywords
Distributed hash tables, Peer to peer, Forward routing tree.

1. INTRODUCTION

Many peer-to-peer (P2P) systems are based on distributed hash tables (DHTs), using a hash table-like interface to publish and lookup objects. DHT based P2P systems have proven to be scalable, robust, efficient and generally applicable. As a result, DHT has become a general infrastructure for building many P2P distributed applications such as archive storage systems, data management systems, application-level multicasts and discovery services. It is a difficult task to implement, deploy and manage a full-edged DHT for an application, it is valuable to take the DHT infrastructure and related services as an application-independent building block to implement certain key components for many applications. The DHT can act as an outsourced service for easing the implementation of applications, which can inherit the scalability and robustness of the DHT. As the capability of the existing DHT is limited, many services may be built on the same DHT infrastructure to support different applications and these services should be designed without modifying the underlying DHT. The basic capability supported by the DHT infrastructure is to exact-match query. However, the ever-wider use of DHT infrastructures has found applications that require support for range query. Examples of range query include the query "10,000 < salary < 20,000" in P2P data management systems, the query in

grid information services and queries for game information in an area in P2P online games. A number of range query schemes have been proposed for DHT-based P2P systems. One important category of them is the general range query scheme which is built entirely on top of existing DHT infrastructures and does not need to modify the topology or behavior of the underlying DHTs.

This way of using DHTs as a shared general infrastructure allows the different applications to be built on the same DHT infrastructure, providing the range query capability without the cost of specifically tuning the underlying DHT. However, such schemes do not adapt the behavior of the underlying DHT to the requirement of range queries; often they are not very efficient. In most existing general range query schemes, the query delay depends on both the total number of peers in the system (N) and the size of the query space or the specific query. As a result, these schemes cannot guarantee to return all query results in a bounded delay that is related only to the scale of the system. When the whole query space or the queried range is large, the query execution can be very slow. This paper describes the Armada, an efficient, delay bounded general range query scheme. Armada operates on top of FissionE, Figure 1 A high-performance constant-degree DHT scheme and does not need to modify the underlying FissionE infrastructure.

Armada provides support for efficient single-attribute and multiple-attribute range queries and can return all query results in a bounded delay, independent of the size of the query space or the queried range.

The main contributions of this paper include the following: The partition tree model to provide order preserving mappings from the query space to the namespace of FissionE. The single-attribute naming algorithm Single hash the multiple-attribute naming algorithm and Multiple hash are designed to assign adjoining ObjectIDs to objects with close attribute values, so that they can be published to the same or related peers in the system to support the efficient range queries. The forward routing tree (FRT) is matches the search paths of range queries to the underlying FissionE topology efficiently. Based on the tree, the range query processing algorithms: PrunIng Routing Algorithm (PIRA) and Multiple- attribute prunIng Routing Algorithm (MIRA) are used to perform single-attribute and multiple attribute range queries within a bounded delay.

Analyze the lower bounds of the delay range of queries, message cost evaluate the query delay and message cost of Armada by both analysis and simulations. [1] I. Stoica, R. Morris, D. Liben-Nowell, D.R. Karger, M.F. Kaashoek, F. Dabek, and H. Balakrishnan have analyses the scalable processing of data in peer

to peer internet applications. [2] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker have analyses the peer of content
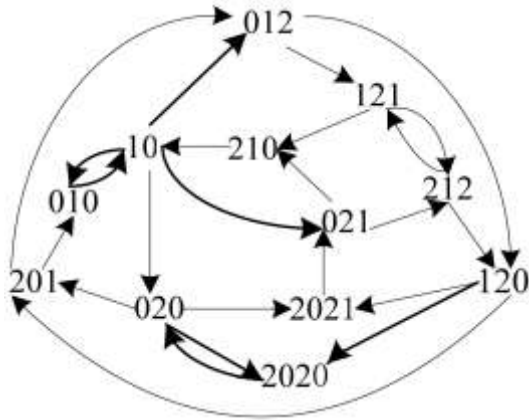

Figure 1. FissionE overlay

in the addressable network. A. Rowstron and P. Drusch have suggested the Distributed Object Location [3] and Routing forLarge-Scale Peer-to-Peer Systems. K.C. Zatloukal and N.J.A. Harvey have analyses an [4] Optimal Congestion, Locality, Degree, and Search Time. P. Ganesan, B. Yang, and H. Garcia-

Molina have [5-7] analyses multidimensional queries in P2P Systems. J.A. Orenstein and T.H. Merrett have [8] analyses class of data structures for Associative Searching. G.S. Manku suggested Balanced [9] Binary Trees for ID Management and Load Balance in Distributed Hash Tables.

2. THEORETICAL ANALYSIS

FissionE provides support for scalable and efficient exact-match query of distributed objects on peers. However, it is difficult to support range queries for attribute values. This paper has design a general range query scheme, called Armada to support range queries on top of FissionE without modifying the underlying FissionE DHT. The basic components of Armada are order-preserving naming and range query processing. Previously Armada first uses order-preserving naming algorithms to assign the objects with close attribute values based on the number of attributes that queried. Range of queries can be classified as the single-attribute range query and the multiple-attribute range query. Armada adopts order-preserving naming algorithms such as Single hash, multiple hashes, query processing algorithms PIRA and MIRA, to perform single-attribute and multiple-attribute range queries, respectively. The components of Armada and its relation to the underlying FissionE DHT are shown in Figure 2.
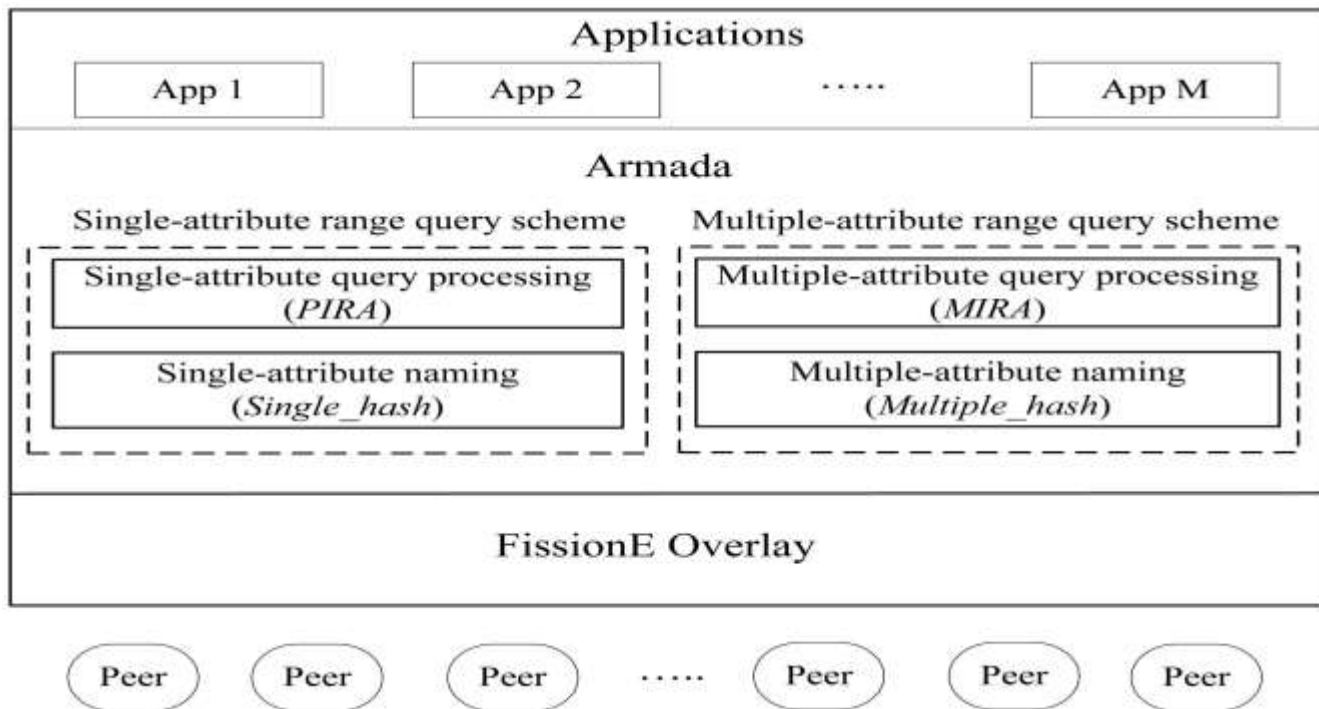

Figure 2. Armada's component

2.1 Single-Attribute Range Query

The proposed model of partition tree is used to design the single hash algorithm. The structure of the partition tree is similar to complete binary tree, but different labels of edges and branches of the root. The partition tree levels with the root node at the 0th level. The root node has three child nodes. The intermediate nodes have only two children. Labels of edges from a father node to its children can be 0/1/2, increasing from left to right, but there should be different from in-edge's label of the father node. The label of the subsequent node is from the path of the concatenation of the label edges.

2.2 Multiple-Attribute Range Query

The naming algorithm & multiple hashes can be designed based on the partition tree of multiple – attributes space.

For any object O with the multiple attribute value $V \leq v_0, v_1, \ldots v_{m-1} >$, V is surely in a subspace represented by a leaf node in the partition tree. Then, the label of the leaf node is assigned as O's ObjectID. The idea of the multiple hash algorithms is similar to the Single hash algorithm.

3. DATA FLOW DIAGRAM

A data-flow diagram (DFD) is a graphical representation of the "flow" of data through an information system. DFDs can also be used for the visualization of data processing. The data items flow from an external data source / an internal data store to an internal data store / an external data sink, through an internal process is called DFD. Figure 3 Shows the flow chart for the processing of query.
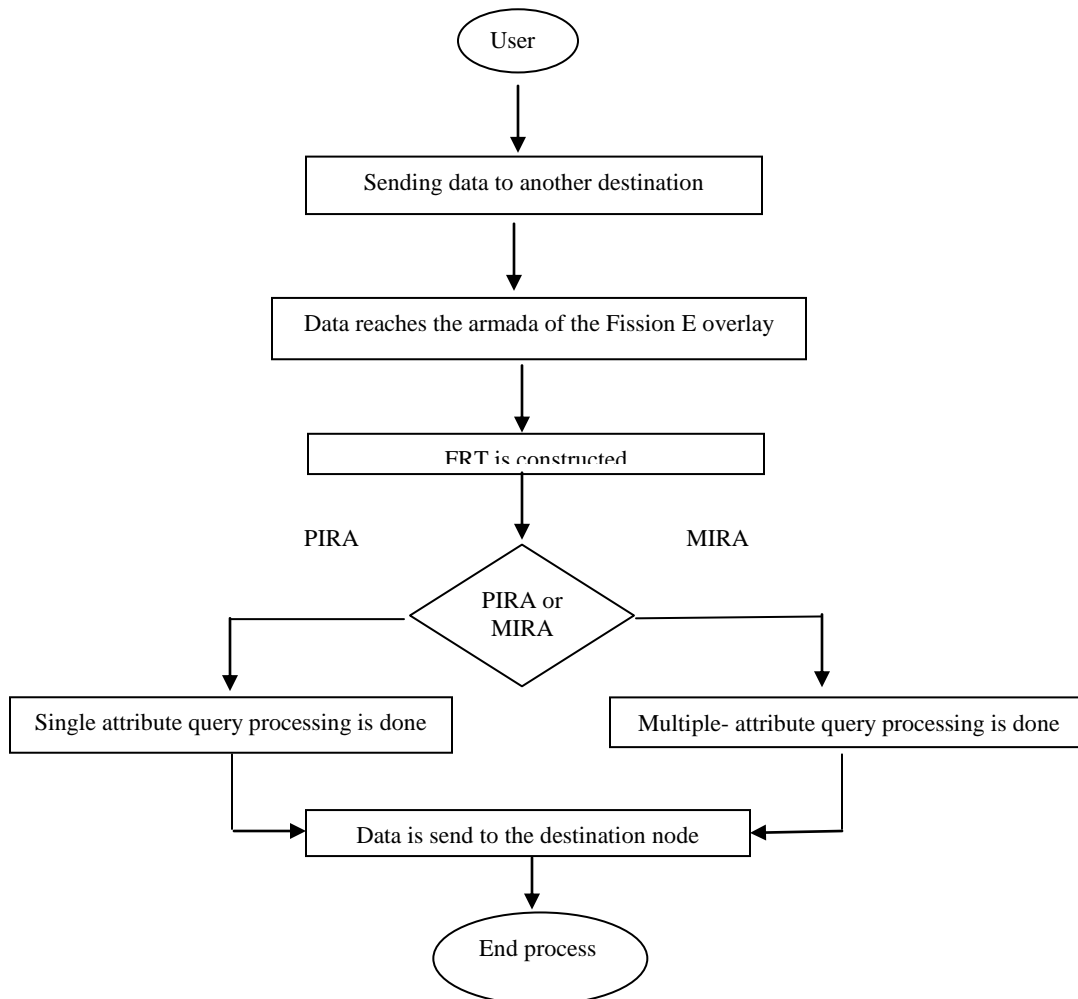

Figure 3. Flow chart for processing of query

A DFD will not provides the information about the timing or ordering of processes or whether the processes will operate in sequence or in parallel. It is therefore quite different from a flowchart which shows the flow of control through an algorithm allowing a reader to determine the operations performed order & circumstances. But it will not shows the input and output data types. It will indicates the origin of input and output data stored.

4. SAMPLE SYSTEM

Range query schemes for DHT-based systems can be categorized in general or customized schemes. General range query schemes are entirely layered on top of the existing DHTs and do not modify the underlying DHT. Customized schemes either makes use of custom-designed P2P overlays or add specific modifications to the behavior of the underlying DHTs. This paper focus on general range query schemes.

4.1 Simulation Result & Analysis

The simulation results shows that Armada can achieve high efficiency in query processing. For any single-attribute or multiple-attribute range query, Armada can return all query results within 2 logN. Figure 4 & 5 show the effects of sample peer Vs error percentage & effects of skew Vs sample size.

The Armada average message cost of single-attribute range queries is about $logN + 2N – 2$, n is the number of peers that intersect with the query which is very close to its asymptotic lower bound. Figure 6 & 7 shows the effects of clustering Vs error percentage & errcts of clustering Vs sample size. Armada uses FissionE as the underlying DHT to organize the peers in an overlay and to deal with the dynamic join or leave of peers. The average degree of the FissionE overlay and its average routing delay is less than logN. Armada is the first delay-bounded range query scheme on top of constant-degree DHTs. Figure 8 Shows the sample output for the processing query in the network
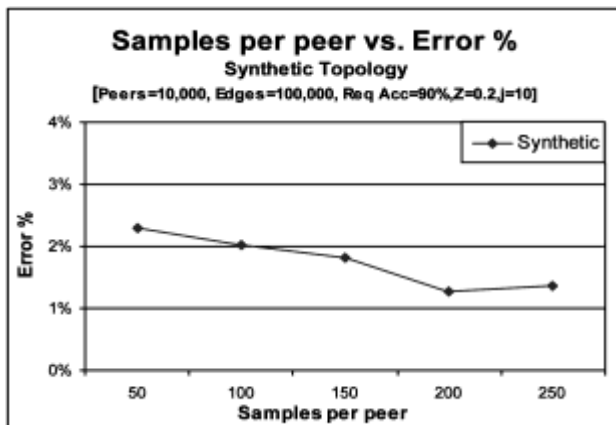

Figure 5. Effects of Skew Vs Sample Size

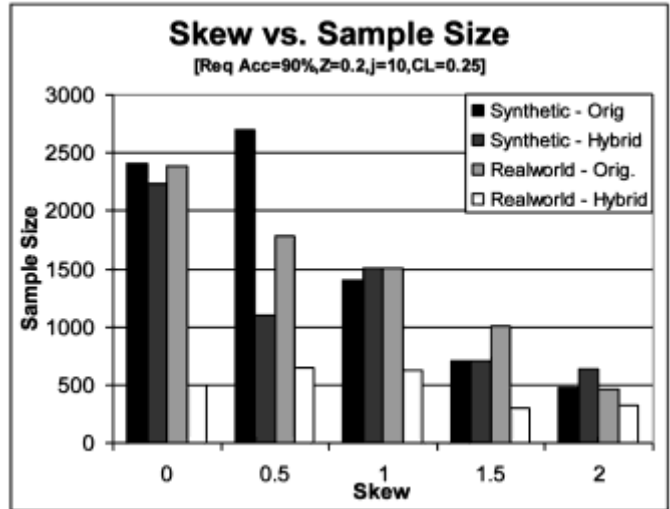**Figure 6. Effects of Clustering Vs. Error Percentage**

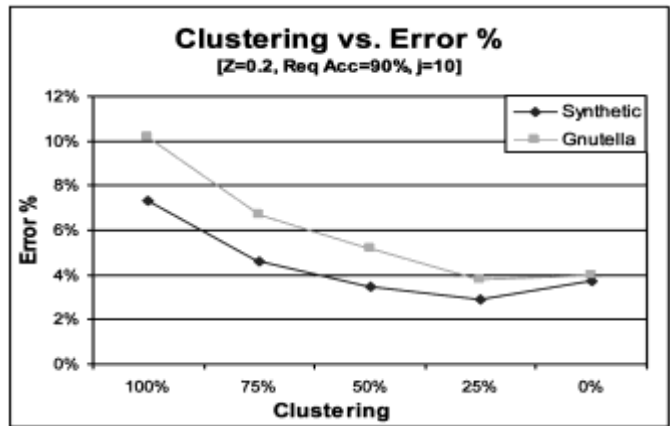**Figure 4. Effects of sample Peer Vs Error Percentage**
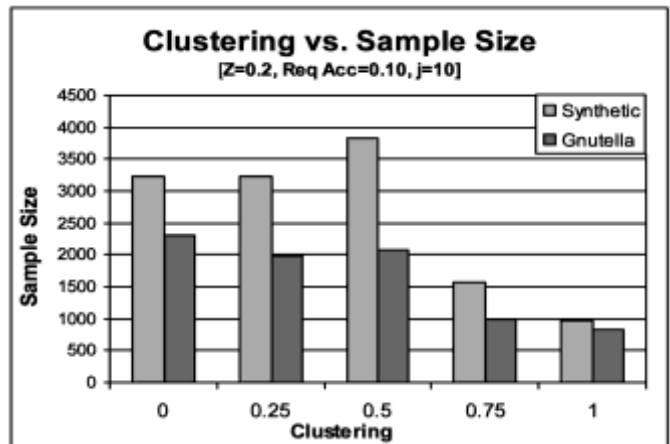

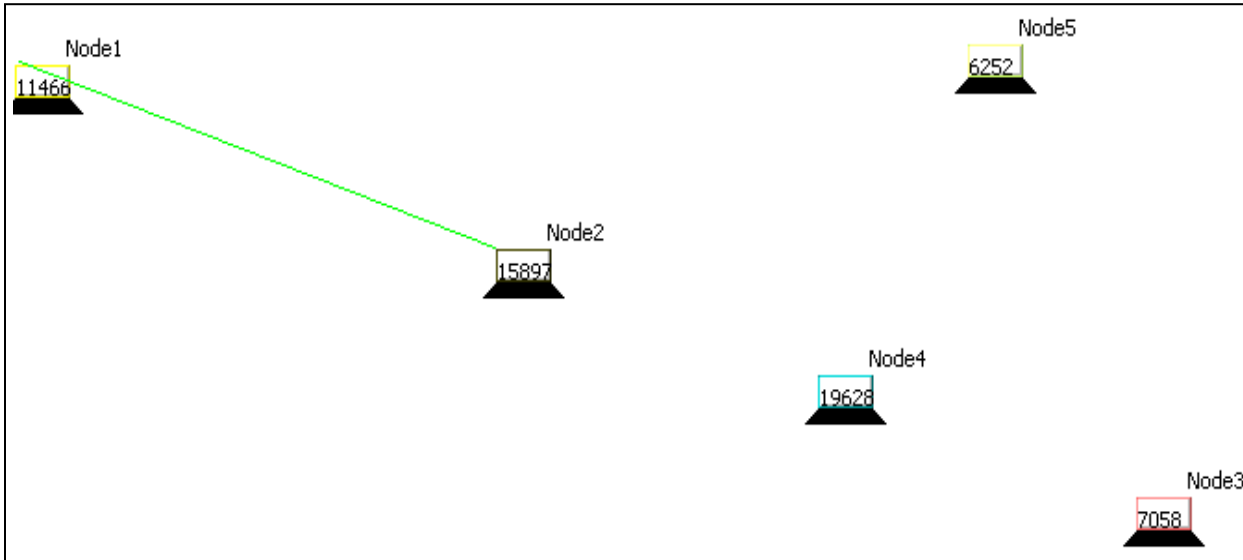**Figure7.Effects of Clustering Vs. Sample Size**

Figure 8: Processing of Query in Networks

5. CONCLUSION

The proposed delay-bounded general range query scheme built on top of FissionE which is a high-performance constant-degree DHT Armada. It supports single-attribute and multiple attribute range queries. Both analytical and simulation results demonstrated the delay-bounded and highly efficient. The average query delay is less than logN and the maximum delay is less than 2 logN, independent of the size of query space and specific queries. The average message cost of single-attribute queries is logN + 2n - 2 where n = number of peers that intersect with the query, which is very close to the lower bound on message cost of range queries on constant-degree DHTs. Armada has been used for resource discovery in the system. The future work includes improving the runtime load balancing of Armada and relieving query hotspots. The extending of Armada to support attribute values in various forms, provide other complex query capabilities such as the top-k query and fuzzy query.

ACKNOWLEDGEMENT

REFERENCES

[1] I. Stoica, R. Morris, D. Liben-Nowell, D.R. Karger, M.F. Kaashoek, F. Dabek, and H.   Balakrishnan, "Chord: A Scalable Peer-to-Peer Lookup Protocol for Internet Applications," IEEE/ACM Trans. Networking, vol. 11, no. 1, pp. 17-32, Feb. 2003.

[2] S. Ratnasamy, P. Francis, M. Handley, R. Karp, and S. Shenker, "A Scalable Content-Addressable Network," Proc. ACM SIGCOMM '01, pp. 149-160, 2001.

[3] A. Rowstron and P. Druschel, "Pastry: Scalable, Distributed Object Location and Routing for Large-Scale Peer-to-Peer Systems," Proc. IFIP/ACM Int'l Conf. Distributed Systems Platforms (Middleware '01), pp. 329-350, Nov. 2001.

[4] K.C. Zatloukal and N.J.A. Harvey, "Family Trees: An Ordered Dictionary with Optimal Congestion, Locality, Degree, and Search Time," Proc. 15th Ann. ACM-SIAM Symp. Discrete Algorithms (SODA '04), pp. 301-310, 2004.

[5] P. Ganesan, B. Yang, and H. Garcia-Molina, "One Torus to Rule Them All: Multidimensional Queries in P2P Systems," Proc. Seventh Int'l Workshop Web and Databases (WebDB '04), June 2004.

[6] B. Liu, W.C. Lee, and D.L. Lee, "Supporting Complex Multi- Dimensional Queries in P2P Systems," Proc. 25th Int'l Conf. Distributed Computing Systems (ICDCS), 2005.

[7] H.V. Jagadish, B.C. Ooi, Q.H. Vu, R. Zhang, and A. Zhou, "VBI-Tree: A Peer-to-Peer Framework for Supporting Multi- Dimensional Indexing Schemes," Proc. 22nd IEEE Int'l Conf.Data Eng. (ICDE), 2006.

[8] J.A. Orenstein and T.H. Merrett, "A Class of Data Structures for Associative Searching," Proc. Third ACM SIGACT-SIGMOD Symp. Principles of Database Systems (PODS), 1984.

[9] G.S. Manku, "Balanced Binary Trees for ID Management and Load Balance in Distributed Hash Tables," Proc. 23rd ACM Symp. Principles of Distributed Computing (PODC '04), pp. 197-205, June 2004.