# CS 654 – Distributed Systems
Project Proposal
W Anthony Young - 20161423

Database systems have been in use for many years as a means to store information. Large organizations store records related to employees, customers and products, while healthcare professionals and researchers store information related to patients and research projects. Databases have the power to organize our information, and provide a quick and easy way to access it.

For the past several years, Distributed Databases have entered the information storage arena. They allow for local autonomy, improved query performance, improved reliability of data and availability of access, and a high level of expandability and easy data sharing [9]. Distributed Databases provide an organization the flexibility to tune storage and access protocols to suit their infrastructure. For example, data that is used frequently can be replicated many times to speed up access. As well, data can be spread amongst many servers to increase redundancy.

Recently, Peer-to-Peer architecture has been employed in Database systems. Peer-to-Peer architecture allows a system to act as both the client (for performing queries and interacting with a user) as well as a server (to provide results to queries posed by other clients on the network) [8].

Peer-to-Peer Databases pose many interesting implementation challenges. They are also fundamentally different from Distributed Databases in several key ways [7]:

- Nodes may join and leave a Peer-to-Peer network at any time: In Distributed Databases, nodes are added out of necessity (ie: for redundancy or growth) and are known to the cluster ahead of time.
- The schema for a Peer-to-Peer Database is not global: In Distributed Databases, the schema is standardized across each node. In a Peer-to-Peer Database, there may be several schemas used to represent the same data on different nodes.
- The data in a Peer-to-Peer Database might not be complete: Distributed Databases contain a complete set of information in each cluster. However, a group of Peer-to-Peer systems might not have the complete set of information required to accurately and completely answer a query.
- Queries in Peer-to-Peer Databases must be routed to many nodes: In Distributed Databases, the query can be routed to a relatively small set of nodes. In Peer-to-Peer Databases, the query must be passed to many nodes in order to return an accurate result set.

There are several key issues of importance to Peer-to-Peer Database systems. They help govern how well Peer-to-Peer Database systems function:

- Scalability: A system must be able to handle an ever-increasing community of users [10].
- Availability: Nodes should be able to communicate with, and receive data from, each other. As well, data should be replicated at, and retrievable from, several sources [4].

- Performance: The network should return results with the smallest latency possible, and communication between nodes should be as efficient as possible [11].
- Data Authenticity: Can a node tell the difference between a correct and an incorrect query hit? It is obvious that incorrect hits should be avoided [4].
- Security: Ensuring that authorized users are the only ones who may make use of privileged data [3].

During this project, the above key issues in Peer-to-Peer systems will be discussed. As well, currently proposed systems for Peer-to-Peer Databases will be evaluated according to each of the key issues. It is hoped that a critical insight into how such systems can be improved will be gained from this in-depth analysis.

The paper will begin with an overview of Peer-to-Peer Database systems, their characteristics, applications, strengths and weaknesses. Following the overview, critical analysis of APPOINT [2], DBGlobe [1], PeerDB [7], Edutella [6], and PDP [5], will take place. A conclusion will follow detailing some suggestions by the author regarding methods to improve such systems.

References:

1. S. Abiteboul, D. Pfoser, E. Pitoura, G. Samaras, and M. Vazirgiannis. DBGlobe: A Service-Oriented P2P System for Global Computing. In ACM SIGMOD Record 32(3), September 2003.
2. F. Brabec, H. Samet, and E. Tanin. Remote Access to Large Spatial Databases. In Proceedings of the Tenth ACM International Symposium on Advances in Geographic Information Systems.
3. M. Ciglaric, and T. Vidmar. Management of peer-to-peer systems. In Proceedings of the Parallel and Distributed Processing Symposium, April 2003.
4. N. Daswani, H. Garcia-Molina, and B. Yang. Open problems in data-sharing peer-to-peer systems. In 9th International Conference on Database Theory, January 2003.
5. W. Hoschek. A Unified Peer-to-Peer Database Protocol. Proceedings of the International IEEE/ACM Workshop on Grid Computing, November 2002.
6. W. Nejdl, W. Siberski, and M. Sintek. Design Issues and Challenges for RDF- and Schema-Based Peer-to-Peer Systems. In ACM SIGMOD Record 32(3), September 2003.
7. W. S. Ng, B. C. Ooi, K. L. Tan, and A.Y. Zhou. PeerDB: A P2P-based system for Distributed Data Sharing. In International Conference on Data Engineering (ICDE), 2003.
8. B. C. Ooi, Y. Shu, and K. L. Tan. Relational data sharing in peer-based data management systems. In ACM SIGMOD Record 32(3), September 2003.
9. Özsu, M.T. and Valduriez, P. 1991. Principles of Distributed Database Systems. Prentice Hall, Englewood Cliffs, NJ.
10. Scalability Issues in Large Peer-to-Peer Networks - http://www.ececs.uc.edu/~mjovanov/Research/paper.html.
11. X. Zhang, J. Liu, Q. Zhang, and W. Zhu. gMeasure: a group-based network performance measurement service for peer-to-peer applications. In Proceedings of the Global Telecommunications Conference, 2002.