Unit 2 Connection Relationships

**Corey Crooks**

**Purdue University Global**

**IT332 – Principles of Information Systems Architecture**

**Jonathan Abramson**

**July 19th, 2023**

In the digital age, there are a great many ways to pass information around. This exchange is vital for the daily operations of even today’s largest businesses. With that in mind, it’s important to choose the right method of data sharing when implementing a network’s functionality. These methods come in two popular forms, although there are many others; peer-to-peer and client/server relationships.

Peer-to-Peer is a method of data exchange that highlights the low-cost of simple and minified network infrastructure. In this relationship, the sender and receiver both share the same precedence (University of Texas at Austin IT Support, 2015). Due to this, data is simply sent and received with no intervening factors. This eliminates the need for a costly network infrastructure to be placed around the globe where you plan on doing business, and instead uses the individual connections of both peers in order to conduct the information exchange. Now, this method does indeed have downsides. For example, this data is extremely susceptible to malicious tampering. Since there is not intervening blocker, bad data can be sent by one peer with no way to validate on the second other than costly routines that will use up vital processing resources on each and every client. This also highlights the faults of individual peer connection and resources. For example, if one device significantly lacks in processing capability, the entire exchange will be slowed, since they are also required for the “heartbeat” of the exchange.

An alternative to peer-to-peer connections would be Client/Server relationships. In this routine, all devices act similarly to how they would in peer-to-peer, but instead interact mainly with the server as a client instead of directly with other clients. This sees an overarching server controlling the communications from each client directly (Florida Center for Instructional Technology, 2013). Because of this, data security can be maintained and preserved, as malicious attempts at tampering data can be denied from code running on the server specifically while the individual clients have no control. This can prevent unauthorized access to information or other systems due to a malicious client attack. Another benefit from this approach would be the overhead cost of each client. Should one client lack the resources to properly handle the exchange, such as a slow internet speed or improper connection, they may simply be dropped from the exchange without disrupting the other clients and needing to add additional logic to ignore the problem participant. One downside of this exchange method, however, would be the overall cost of such a setup. The connection that a client has to the server is absolutely critical to the success of the informational exchange happening (Wood, n.d.). To accommodate this, you would need to anticipate where your clients will be located. In these locations, it is important to have datacenters nearby. The proximity of datacenters will ensure each client may communicate effectively and without delay. This will help your entire exchange run more smoothly, and also minimize data errors that may come from a slow connection such as data overlapping.

Which data exchange method to use is a complicated question to answer. There is no “right” answer when it comes to network architecture, as each solution presents different issues and benefits. However, it has come to my understanding that this company is not limited in its resources to provide quality connections with network size, requirements, or hardware. With this information, I would recommend implementing a wide network of servers for a Client/Server relationship. Peer-to-Peer relationships are much more unreliable compared to a Client/Server connection, and mainly exist to limit the financial obligation that it would take to establish a global network of servers. Client/Server relationships provide the backbone of reliability and speed that is necessary for this company to grow, and serve a complex grouping of markets in the informational age of today.

It is important to note that this is only a suggestion, and may not take into account certain factors such as employee size, team requirements, or software considerations. All of these are important factors when implementing both solutions. A client/server relationship will only work if there is a skilled team to implement the approach fit with designations and error handling. Be sure to take this information into account when deciding on which approach is best for the company.

# **References**

Florida Center for Instructional Technology. (2013). *What is a Network Operating System?* Retrieved from fcit.usf.edu: https://fcit.usf.edu/network/chap6/chap6.htm

University of Texas at Austin IT Support. (2015, September 14). *Dangers of Peer-to-Peer Networking*. Retrieved from cns.utexas.edu: https://sites.cns.utexas.edu/oit-blog/blog/dangers-peer-peer-networking

Wood, C. (n.d.). *Client/Server Data Serving for High Performance Computing*. Retrieved from Maximum Strategy Incorporated: https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=08c406f5ac2057129957d8189f412347613e73bd