

Research Journal

COMP30 - Research Journal

1607804

April 3, 2017

1

The first paper in this journal [1] provides a method of overcoming latency [2] issues stemming from overcrowded online environments. The paper starts by discussing ways in which software engineers have handled latency issues in the past. On such method is done by predicting the position of the player by knowing the current position, the movement speed of the player, the delta time and the previous position[3]. Mathematically described:

$$P_c = V * \Delta t + P_p \quad (1)$$

This is more commonly known as dead-reckoning. Another method described in this paper that is used in games is to subdivide a crowded world into sub-regions called cells. Synchronisation is only between objects that are located in the same cell or "locale"[4]. Examples of this can be seen in the online game RuneScape[5]. In addition to this, I believe they change the size of these cells depending on how crowded the area is. The author then proceeds to talk about the M-COVE model, this model at a high level is as follows: Does the client have recent detailed orders? If so employ dead reckoning. If not ask is the agent out of the clients view? If so Warp to the destination. If not ask is the

agent in a high waiting area? If so wait. If not move. From experiments done by the author the M-COVE model compared with traditional synchronisation they were able to get 740 more clients in the same environment with exactly the same bandwidth.

The second paper in this article introduces a way of networking without the use of authoritative servers, through the use of peer-to-peer networks[6]. Creating an online game in this way is advantageous for companies that do not want to incur the costs of server maintenance. In this paper, similar to the first, they mention this idea of 'zones', where all the global states of a specific area are bound to that areas zone data. They also mention how player data can be bound in these zones when the player is in that specific area. They go on to talk about "zone owners" the client responsible for that zone. When the client becomes the owner it also becomes the authority of that zone. This means that any conflicts that occur in the zone will be resolved dependent on the zone owners state. Using a peer-to-peer network may require less resources but it also increases latency. The latency will be dependent on the zone owners connection to other clients which may be unreliable. An example of this is with the old Call Of Duty games. They were peer-to-peer. At the beginning of the game, one client was chosen to be the host being responsible for the global states. If the hosts bandwidth fell, the other clients would experience an increase in their latency. This means the other players were being correct to the hosts game state as the host was the authoritative figure. See "Lag Switches"[7] for more details on how this was abused. So while the Zoned federation model proposed in this paper may be beneficial in some cases it increases the chances for exploitation.

In this next paper[8] the author analyses the client-server and the peer-to-peer architectures from two main viewpoints: Bandwith usage and inconsistency resolution latency. The author starts by saying that "players often use limited-bandwidth" and goes on to say some may use dialup. While this may have been a stronger point back when this paper was published(2003), the bandwidth nowadays far exceeds the bandwidth found in 2003.

One interesting concept this paper discusses is that of inconsistency resolution latency, this is the time it takes for two nodes with inconsistent states to resolve. The author also suggests a new architecture called peer to peer with a central arbiter or (PP-CA). In this architecture players exchange game state updates like they do in the peer to peer model. They also send updates to the central arbiter, the CA's role is to simulate a game state from the updates it receives and detects any inconsistencies with the clients. The CA will not send any updates until an inconsistency is found, at that time it will send an update to correct it, state consistency is paramount in online games[9]. Even though other researchers have found PP-CA architecture to produce "good latency, good administrative control and good scalability" [10, p.22] also stating that it is a "good match for any real-time game" [10, p.22] examples of this actually being implemented in the games industry are non-existent.

The two transport protocols most commonly used are UDP and TCP[11]. The authors of this next paper [12] go into depth on how a Wide area network (WAN) which is conveying both UDP and TCP traffic at the same time makes UDP packet loss more likely to occur. The introduction highlights that UDP meets the requirements better for time sensitive applications, applications that have their own control flow and transmission mechanisms. TCP, on the other hand, is not so good for time sensitive applications, this is because if one packet is dropped it has to be re-sent before the sent information can be read. UDP packets are unordered whereas TCP packets are in the order that they are sent. The results of their experiment show that UDP streams suffer from TCP synchronisation. In addition to this, they recommend using UDP as the transport protocol when the application is time sensitive for example, a first-person shooter. They also recommend using small packet sizes with a high transmission rate to see the best results. Taking this forward into my own work, if I ever find myself needing to implement a time sensitive game I will use UDP for the gameplay but for parts that depend more heavily on the order for example voice chat I will TCP as the transport

layer.

References

- [1] J. Parker and N. Sorenson, “A novel network architecture for crowded online environments,” in *Proceedings of the 2008 ACM SIGGRAPH symposium on Video games*. ACM, 2008, pp. 129–134.
- [2] M. Claypool and K. Claypool, “Latency and player actions in online games,” *Communications of the ACM*, vol. 49, no. 11, pp. 40–45, 2006.
- [3] S. Li, C. Chen, and L. Li, “A method of using personal habits for path prediction in network games,” in *Data, Privacy, and E-Commerce, 2007. ISDPE 2007. The First International Symposium on*. IEEE, 2007, pp. 4143–4143.
- [4] J. W. Barrus, R. C. Waters, and D. B. Anderson, “Locales and beacons: Efficient and precise support for large multi-user virtual environments,” in *Virtual Reality Annual International Symposium, 1996., Proceedings of the IEEE 1996*. IEEE, 1996, pp. 204–213.
- [5] Runescape. [Online]. Available: <http://www.runescape.com/community>
- [6] T. Iimura, H. Hazeyama, and Y. Kadobayashi, “Zoned federation of game servers: a peer-to-peer approach to scalable multi-player online games,” in *Proceedings of 3rd ACM SIGCOMM workshop on Network and system support for games*. ACM, 2004, pp. 116–120.
- [7] B. Mitchell. What is a lag switch? [Online]. Available: <https://www.lifewire.com/what-is-a-lag-switch-817481>
- [8] J. D. Pellegrino and C. Dovrolis, “Bandwidth requirement and state consistency

- in three multiplayer game architectures,” in *Proceedings of the 2nd workshop on Network and system support for games*. ACM, 2003, pp. 52–59.
- [9] L. Pantel and L. C. Wolf, “On the impact of delay on real-time multiplayer games,” in *Proceedings of the 12th international workshop on Network and operating systems support for digital audio and video*. ACM, 2002, pp. 23–29.
- [10] P. Sutinrerk, “Mirrored arbiter architecture: a network architecture for large scale multiplayer game,” Ph.D. dissertation, California State Polytechnic University, Pomona, 2006.
- [11] S. Bishop, M. Fairbairn, M. Norrish, P. Sewell, M. Smith, and K. Wansbrough, “Rigorous specification and conformance testing techniques for network protocols, as applied to tcp, udp, and sockets,” in *ACM SIGCOMM Computer Communication Review*, vol. 35, no. 4. ACM, 2005, pp. 265–276.
- [12] S. Hidenari, Y. Hori, and H. Sunahara, “Characteristics of udp packet loss: Effect of tcp traffic,” in *Proceeding of the 7th Annual Conference of the Internet Society*, June, 1997, pp. 24–27.