# End of year progress report 2011

## John Gilmore

### What I’m working on:

Massively multiplayer online games have gained much popularity in recent years. The problem is that they’re still very expensive to develop and run. I’m working on peer-to-peer MMOGs, also called peer-to-peer massively multiuser virtual environments (P2P MMVEs), which would remove the need for a server, which will in turn save a lot of money for the operator and also allow independent game developers to create MMOGs, without the need for large initial infrastructure investments.

There are other architectures out there that propose to achieve the same goals. None of these architectures are, however, complete. Each of the architectures has some aspect still outstanding and none have really made it to being a commercial product.

The architecture we’re developing attempts to further improve on the peer-to-peer paradigm for gaming, by enabling more responsive games. For my part in this architecture, I’ve decided to focus specifically on how data is stored in a distributed fashion, to still ensure low latency interactions.

Data are stored in two tiers in P2P MMVEs. Game objects are stored in primary memory where clients access them directly and the same objects should also be stored in secondary memory to ensure persistency state, even if the client disconnects from the network. Primary memory and secondary memory storage are called state management and state persistency respectively.

### What I’ve done so far this year:

I’ve made the required changes to have my journal paper accepted by the IEEE transactions on parallel and distributed systems. The paper has been accepted for publication and a pre-published version is available online.

The Badumna architecture was also tested and deployed locally. The knowledge gained from this system has allowed for a better understanding of what the network architecture requires of the state persistency layer. Badumna might still later provide the game that will be used to drive Pithos, our P2P MMVE storage solution.

The design of Pithos required the identification of player groups, which provide us with our multi-tiered system. Developing a grouping algorithm that will identify player groups from the player traces is one of our goals.

We were unable to obtain any player traces from game companies. Our solution has been to generate our own player traces, from a final year engineering project, to be used in the further design of Pithos. The player tracking software has been completed and this will later be used to provide the traces which my grouping algorithm will operate on.

I have implemented a significant part (more than 8000 lines of code) of the Pithos architecture as a simulation. Simulation before implementation allows for greater control over all variables and improved gathering of network wide statistics. We are also able to simulate much larger networks, than would be impossible with a real-world implementation.

During the design of Pithos and further research, I identified a larger structure that has always been hinted at in literature, but never completely defined. This structure contains the state management and state persistency models and is contained within the still larger P2P MMOG model. This model is called the state consistency model and is described in detail in our survey paper. The simulation environment has also been expanded to include the state consistency model and Pithos has been implemented within this environment.

I’ve integrated an existing robust overlay storage into Pithos, which allows me to focus on the overall design and improving the group storage and interaction between the group and overlay storage.

I’ve also started to collect performance statistics from Pithos, based on the metrics defined in our IEEE survey paper. Two of the metrics, namely responsiveness and fairness, can currently be measured and are compared with relevant other storage architectures.

The Pithos simulation and initial results has also been published in another conference publication, which was presented at the MMVE workshop, as part of the IEEE HAVE conference in China.

### Publication details:

* John S. Gilmore and Herman A. Engelbrecht, “***Pithos: A State Persistency Architecture for Peer-to-Peer Massively Multiuser Virtual Environments”***, Proceedings of The 4th International Workshop on Massively Multiuser Virtual Environments (MMVE) at the IEEE International Symposium on Haptic Audio-Visual Environments and Games (HAVE 2011), 15 October, Qinhuangdao, China
* John S. Gilmore and Herman A. Engelbrecht, “***A Survey of State Persistency in Peer-to-Peer Massively Multiplayer Online Games”***, IEEE Transactions on Parallel and Distributed Systems, July 29, 2011

### What I plan to do next year:

1. The Pithos grouping algorithm still has to be developed and tested and integrated into the Pithos simulation.
2. Pithos should be extended to support get and modify requests as well as a reliable signalling mechanism to the higher layer.
3. Implement time to live for game objects in group storage.
4. Add mechanisms to handle network churn in group storage.
5. Test the system’s reliability and security for various levels of churn and various numbers of malicious users.
6. Publish one or two conference papers.
7. Write up my PhD dissertation.
8. Submit a journal publication.
9. Submit my PhD dissertation.

### Conferences and journals:

In the next year:

1. I would like to publish a paper at the 2012 MMVE workshop,
2. perhaps publish another paper at the 2012 NETGAMES workshop and
3. submit a paper for publication in the IEEE transactions on parallel and distributed systems journal.

### Tracking progress:

When looking back on the Gantt chart, submitted with the initial project proposal, everything seems to be on track. More time might have been spent on developing a testing framework, but this is difficult if the nature of what needs to be tested is not yet known.

The other items have all been completed, except implementing Badumna in the cloud. After some thought, a cloud implementation might not provide the required data to reliably test the architecture. The reason for this is that a primary metric is latency, but the latencies amongst nodes in the cloud are not comparable with nodes on the Internet.

This issue is being dealt with by using real recorded latency data and creating nodes that poses those latencies in a simulation. The idea is then to eventually run the actual state persistency implementation, but on simulated nodes, with latencies that match what is measured in the real world. Simulation is used, because the number of nodes required (at least 3000) is infeasible to set up in practice.

The focus of the research has also somewhat shifted away from designing and implementing a complete P2P MMOG architecture to designing and implementing a state management and state persistency architecture.

The reason for this is that a P2P MMOG architecture is a very large project, which required a research team of ten or twenty people, working on it for a few years. This is what is being done in other research labs in the rest of the world. My project is, however, begin designed and implemented with the P2P MMOG architecture proposed in my original PhD proposal in mind, but the other modules will have to be implemented by other students.

We are currently in the process of adding students to the project to accelerate development of the larger architecture. These students will have my code base to start from, which has been designed to allow for the incorporation of other P2P MMOG modules.