ABSTRACT
Using peer-to-Peer (P2P) architectures for large scale interactive applications such as Massively Multiplayer Online Games (MMOG) is very challenging because of the difficulties to maintain a consistent game world in a distributed topology and exchange game state information in the P2P network without a central sever. In this demo proposal we present the innovative design and implementation of PartyPeer, an online social game which supports a massive number of users using our P2P based streaming network called ACTIVE+. We also discuss some of the implementation challenges when building this real-world P2P based game.

1. INTRODUCTION
Massively multiplayer online games (MMOG) have become a widely successful online business with revenue in the billions of dollars (2005). The dominant design for current MMOG games is the client-server paradigm. To the best of our knowledge, there is no truly successful implementation of an MMOG game built on a P2P-based design exists, despite the tremendous amount of research investigating P2P architectures.

In this proposal, we describe our design of PartyPeer, a P2P based game that supports a massive number of concurrent players in a social-style environment, the prototype of which is shown in Fig. 1 as a private island. We will discuss how we use our ACTIVE+ protocol to address some of the challenges raised previously: scalability, latency and P2P connectivity.

2. THE DESIGN
2.1 Game Overview
PartyPeer is a social game that provides a virtual space where massive number of players can gather and interact with each other. The

2.2 Game Components
PartyPeer has three major components, the game engine, game world and the p2p network module. Game world is composed of the scripts describe the gaming place, 3D object models and other gaming rules and logics. Game engine runs the game world and renders the graphic. p2p network module provides the information exchanging platform among all players.
We use Torque [1] as the game engine for PartyPeer implementation and ACTIVE+ as the network module. Fig. 2 shows these function modules of current game design.

2.3 Features
We believe PartyPeer is an innovative project because of following features:

2.3.1 Low latency voice chatting for large group
PartyPeer provides voice chatting among all participating players and dynamically maintain a low latency voice link among nearby speakers. In our ACTIVE+ network, players are clustered based on their positions in the game world. By using our ACTIVE+ protocol, hundreds or even thousands of people can join in the same space while minimal central resources are needed to coordinate.

2.3.2 Probability based message exchanging
PartyPeer solves the scalability problem in a P2P network by introducing probability based message forwarding mechanism. However, the system is designed so that the players virtually far from each other will have less probability of exchanging game state information as compared to nearby players. By doing so, the data rate required at each node is bonded by a value independent of the total number of players online. This feature makes PartyPeer a very scalable multiple-player online game.

2.3.3 NAT connectivity issue
One important practical issue that has largely been ignored by the research community when designing the P2P based game architecture is that many players connect to Internet using network address translation (NAT) devices such as home DSL modems or cable modems. These NAT devices give players machines a local IP address that is not accessible from outside world without special handling. We designed and implemented our own version of the whole punching protocol to build connections between machines behind NAT devices. We are now testing PartyPeer by players around the world, including far east and Europe and our protocol is running very well.

3. DEMO PLAN
We are planning to demo our game in a private island setting, where our guests can play paintball with players from around the world. They will be able to chat with each other as well.

We currently have already finished the modeling of the private island and the paintball game script. We have also designed two player models, one male and one female. We are continuously adding new player models to the game at this moment. Fig. 3 is a screenshot of our prototype showing several players playing paintball game outside a beech house in the private island.

4. CONCLUSIONS
In this proposal, we have presented a novel design of a peer-to-peer based massively multiplayer online game called PartyPeer. We used our innovative ACTIVE+ protocol to build a scalable and low latency streaming network. Our initial results indicate that it is feasible to implement a MMOG game using P2P based architecture. We also briefly discussed the current progress of our implementation and the plan for demonstration. We are planning to complete the beta version of PartyPeer by the end of summer 2006.

5. REFERENCES

