

Lab* “PeerBlast: A Peer-to-Peer XBlast” (4-6 People)

Multi-player network games (MNGs) are enormously popular. Currently most MNGs are based on a client/server architecture. A single server maintains the game status and provides a consistent view to all participating clients. Obviously, if the server fails, the game is over.

Peer-to-peer computing is the sharing of computer resources and services by direct exchange between client systems (peers). Following this approach there is no central server. All participating peers are responsible for maintaining connectivity and consistency and for ensuring fault-tolerance. Thus, if a peer fails, life goes on.



In this lab you will develop a peer-to-peer variant of the popular XBlast game (PeerBlast). More precisely there is a running and documented version available from the last term lab; you should use it as the fundament for your work. There are still many aspects to be solved and new features to be implemented: Since PeerBlast is an action game, a good architecture has to provide efficient routing of messages in real-time. Furthermore, the game should scale up to hundreds of simultaneous players, and even if multiple hosts fail concurrently, the other players shouldn't be affected. But how can you guarantee a consistent view on the game state (“Did she really blast me? I'm still alive!”) if there isn't an umpiring server?

References:

- <http://get.to/xblast>

Skills

- Network programming in Java

Contacts

- Keno Albrecht, keno.albrecht@inf.ethz.ch, HRS G3, phone 20896
- Roger Wattenhofer, wattenhofer@inf.ethz.ch, HRS G5, phone 26312

* For the new Major in Distributed Systems students have to complete one of our lab projects.