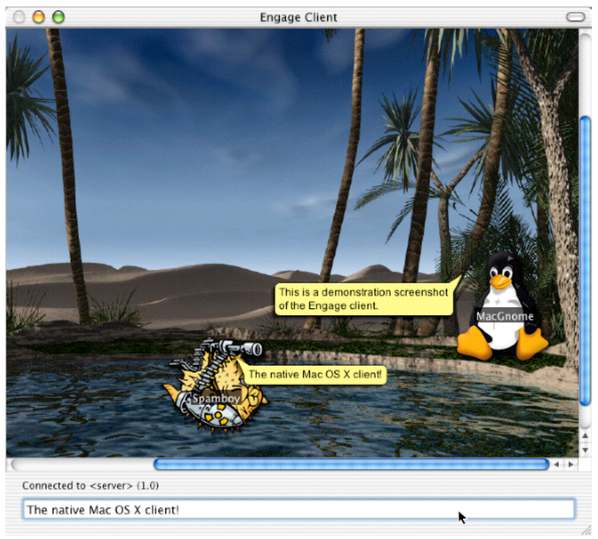
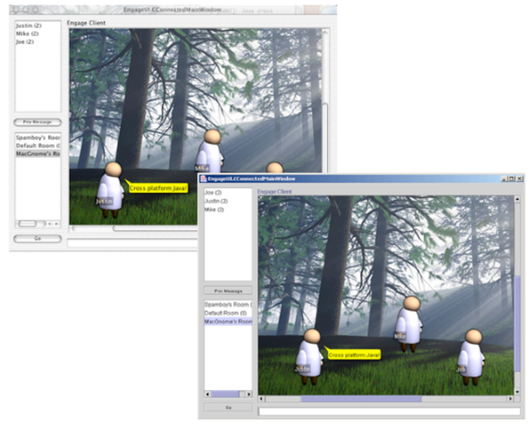


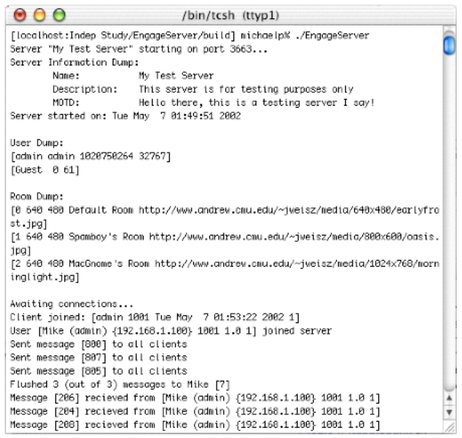
Engage Client

The Engage client showcases the pervasiveness of the graphical metaphor in our communication system. Our Java client ensures cross platform compatibility while our OS X client demonstrates the feasibility of a natively hosted client. Native hosting allows us to take advantage of services not offered in the Java API such as text-to-speech. The Java client source code includes a network layer for interacting with network messages independent of the user interface. This separation of services makes a native interface possible without rewriting network code that doesn't affect the user experience.



Java Clients

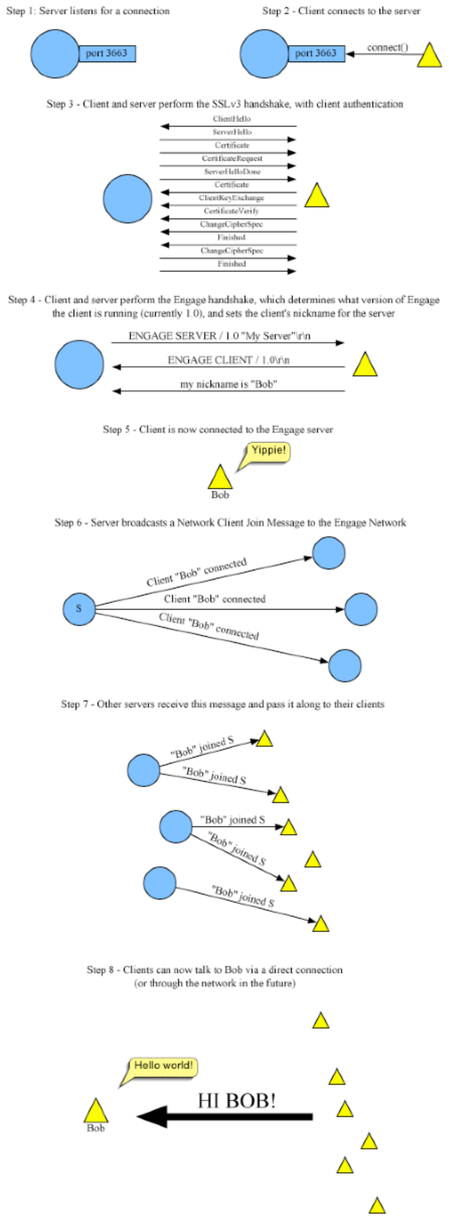
Engage Server



Engage Server (Mac OS X hosted)

The Engage server is the heart of the network structure. It is a POSIX compatible, command line tool intended to handle the routing of both messages between clients connected to just one server and messages between servers on the server peer network. The server peer network maintains redundant lists of online users so each server can route messages to the server where their target is connected.

Servers also employ certificate-based authentication to verify the identity of every user on the network (via SSL/TLS). To this end, some servers may act as certificate authorities that listen for special connection requests. When received, the server signs client certificates using the Engage Network master key. The client or server bearing this certificate is now a trusted member of the Engage Network, and can communicate with other members who have had their keys signed by the same authority.



Future Directions

Application of distributed hash lookup systems for server ↔ server message routing such as Chord from MIT (http://www.pdos.lcs.mit.edu/chord/). This system analyzes an identification number in order to locate a file using O(log n) messages. We can apply this concept to make finding a user more efficient. Currently, redundant active user lists are stored completely on each server node and user login and logout messages must be repeated to all nodes. Chord presents a method of dividing the list storage space and decreasing network traffic.

Enhanced support of media in the client. Currently, only graphical representations of users are permitted. Ideally, we would like to represent any file generically as a world object.