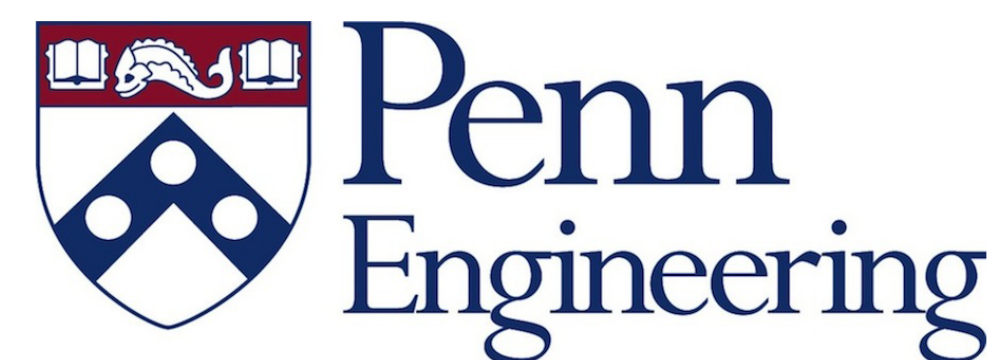


# Tobiko: A WebGL Multiplayer Online Game

Gianni Chen, Advisors: Dr. Norman Badler, Aline Normoyle

Senior Project Poster Day 2013, Department of Computer and Information Science, University of Pennsylvania



## Abstract

Tobiko is a 3D massively multiplayer online game (MMOG) framework using WebGL and WebSockets. It builds some of the baseline features of an MMOG and aims to provide boilerplate code using a new technology stack. On a high level, users connect to a server from their browsers and interact with other users who are also connected at that time. Features can be built on top of this base code to implement functionality specific to a game.

## Goals

Tobiko aims to make these contributions:

- Explores a new alternative for game developers on the web
- Establishes a new front and back end stack for web games and developers
- Provides a base code for future WebGL multiplayer environments

## Introduction

Since the late 1990s, MMOG's have been a popular alternative to other games as well as a pleasant escape from everyday life. More recently, many classic games have been making reimagined appearances as MMOGs in our browsers, few which use WebGL. While WebGL allows for both 2D and 3D rendering, the multiplayer games we currently do see are often in 2D.

## Design

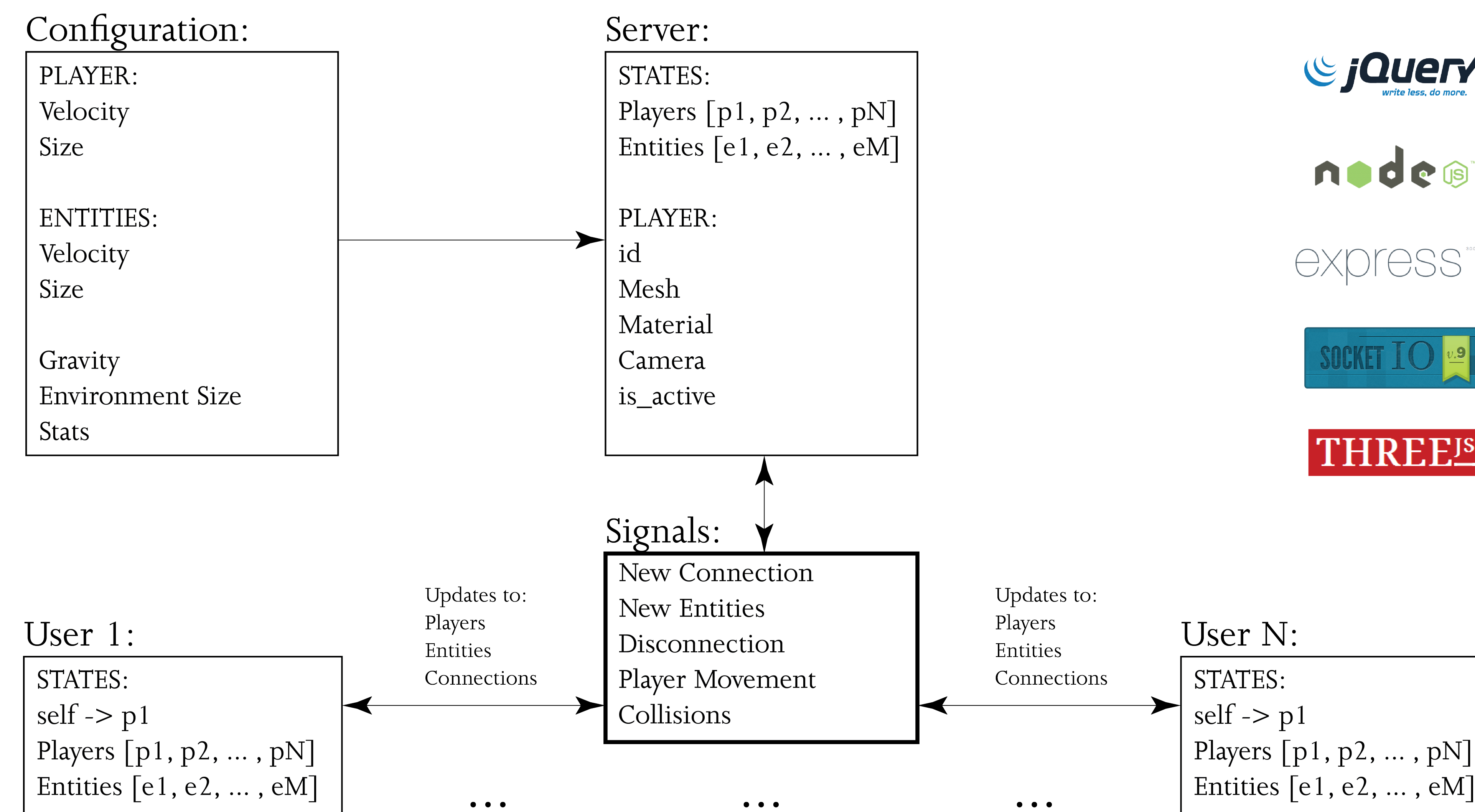


Figure 1: Flow chart of data transfer between clients and server

## In-game Screenshot



Figure 2: Screenshot in game using HexGL's model ship

## Features

- An interactive 3D environment accessible by anyone with a WebGL supported browser
- Support for multiple players to interact in the environment
- A way for players to navigate the environment
- A way for players to eliminate one another (e.g. missiles)
- A competitive way for players to avoid being eliminated (e.g. rolling)
- A presentable and deliverable open source basecode

## Future Work

- Optimize connections and renderings to scale the application for more connections
- Integrate more physically based calculations into player interactions (e.g. pushback from another player's attack)
- Assign object models to players to make for a better demonstration
- Apply more sophisticated renderings of animations (e.g. particle systems, lights)
- Create more detailed terrains so they can be full maps with obstacles

## Acknowledgements

Thank you to Dr. Norm Badler, Aline Normoyle, and MrDoob.