

Doug Koellmer - Programmer



West Hartford, CT
doug@dougkoellmer.com

Open Source



I maintain several projects related to web development, math, physics, games, and application frameworks. Check out my account profiles to the right and my most ambitious endeavours below.

Swarm [source](#), [demo](#), [demo](#)

A next-generation web-based CMS...



...for surfing and organizing large amounts of content in a natural, seamless fashion. Has a meaty HTML5 front-end and fast, scalable Java back-end.

[javascript](#) [java](#) [html5](#) [gwt](#) [gae](#) [bash](#) [git](#)

QuickPhyx [source](#), [demo](#), [forums](#)

A high-level game physics engine...



...with an event-driven DOM-like API, CSS-like property system, multi-language support, and so many other cool features you'd get bored with me listing them.

[haxe](#) [as3](#) [java](#) [javascript](#) [physics](#) [geometry](#) [git](#)

Professional Experience

Most of my paid experience has been in game development, web development, virtual reality, CAD, and related fields. Check out a few select projects to the right and more details below.



Cofounder, CTO

Eagre Interactive
EdTech Startup
Summer 2012 - Recently



- Created the company [website](#).
- Ran a Kickstarter [campaign](#).
- Created a series of interactive HTML5 minigames for McGraw-Hill's Earth Science division. See some samples [here](#) and [here](#).
- Prototyped a radical new kind of [fixed-layout ePub3 reader](#) for textbooks.
- Knee-deep in all the gore that goes along with running a business.

[html5](#) [javascript](#) [php](#) [java](#) [epub3](#) [svn](#) [pulling hair out](#)

Programmer

Venan Entertainment
Mobile Game Studio
Winter 2011 - Spring 2012



- Codeveloped the hit social MMORPG [Book of Heroes](#).
- Codeveloped a social gaming platform running on Amazon's cloud service that serves millions of monthly users.
- Developed an internal web application for Book of Heroes allowing three designers to work concurrently to create hundreds of items, monsters, NPCs, quests, locations, and more.

[ios](#) [android](#) [c++](#) [c](#) [java](#) [php](#) [sql](#) [nosql](#) [aws](#) [svn](#)

Lead Programmer

Johnson Center for Simulation
Serious Game Studio
Spring 2008 - Winter 2011



- Led the programming and design for a [\\$1 million game](#) funded by the Department of Defense to teach metal corrosion principles and prevention. They gave us another \$2 million the next year.
- Led a \$200,000 project funded by [Medrad](#) using force-feedback devices to simulate a gluing procedure that was too expensive to teach with real parts.
- Programmed and co-designed a rhythm-based iOS game called [Touch Tone Hero](#). It currently has a 4/5 rating.
- Created an in-house logging system to track daily project progress and compile reports for clients. It has been used for 4 years and saved countless manager-hours.
- Made a proof-of-concept for using a \$30 Wiimote as a 6-DoF 3d tracker. Funded by the NSF to replace the multi-thousand dollar trackers normally in use.
- Made numerous custom hardware components for the center's various VR systems; mounts, electronics, casings, mock-ups, etc.

[ios](#) [as3](#) [c++](#) [c#](#) [php](#) [aug-reality](#) [positioning-system](#) [svn](#)

Lead Programmer

Fluid Desk
CAD Software Firm
Fall 2005 - Winter 2008



- See a quick presentation of my work [here](#).
- Led a team of three developing a mesh-based geometric primitives API, supporting variably-triangulated, highly-flexible surface approximations, along with operations for creating various sections/intersections thereof.
- Used the above API to construct ~150 distinct 3d piping and ventilation elements, configured dynamically by dimensions retrieved from a UI and manufacturer specifications.
- Developed a Hidden Line Removal (HLR) engine, which takes 3d scenes of elements from any viewpoint and renders a symbolic 2d line drawing of use to sanitary engineers.
- Developed a parametric geometry primitives API as a compliment to the mesh-based API. This was used for HLR as well as extremely accurate collision detection.

[c++](#) [objectarx](#) [autocad](#) [svn](#)