plan.js

Motion Planning for the Web

Clinton Freeman

Movement Toward Web Applications

Web

- increasingly popular
- cross-platform
- no installation
- interpreted languages
- OpenGL ES 2.0

Desktop

- native speeds
- native languages
- full access to system resources
- latest GL/DX features

Reinventing the Native Wheel

- New js libraries
  - three.js - rendering
  - sylvester.js - linear algebra
- No motion planning library
- OMPL for desktop
- Port C/C++ → JS? Yes!*

Open Motion Planning Library

- Many implementations (RRT*, PRM, etc.)
- Integration with ODE, PQP
- Python/OpenGL GUI
- Dependency on Boost

Porting Native Applications

Google Native Client

- sandboxed machine code
- only in Chrome
- very close to native speeds
- security concerns

Mozilla Emscripten

- native to bytecode to javascript
- all browsers
- 2x slowdown
- works today

Emscripten Overview

- Toolchain
  - emcc = gcc
  - emar = ar (etc.)
- Challenges
  - file system
  - threading
- Examples
  - Unity 5, Unreal Engine 4
  - [ammo.js (Bullet collision detection)](#)

Porting OMPL with Emscripten

- Goal: simple 2D planning demo in browser
- Challenges
  - compiling Boost libraries
  - dependency on boost_thread
- Results
  - compiled all Boost libraries except boost_thread
  - working demo of boost_regex
  - compiled individual OMPL source files