WebGL: OpenGL ES in JavaScript

Working Group in Progress

Targeting Public Draft in October 2009

Introducing 3D to HTML5

- **The HTML5 Canvas Element**
  - Canvas2D already used for powerful graphical applications on the Web
  - var ctx = canvas.getContext("3D");
  - API defined on Canvas element, introducing DOM interfaces

- **Using WebIDL in the Specification**
  - Used for W3C Specifications (e.g. Ajax backbone APIs)

- **Shader Based**
  - Portable Shaders on the Web, subject to validation
  - Potential plan for older architectures that don't support Shaders

- **Simple binding to OpenGL ES2.0**
  - Taking intersection of Desktop GL and OpenGL ES
  - Example:
    - Removing capability to do user-supplied texture IDs
    - (Possibly) Requiring VBOs as only data array mechanism

Shaders on the Web

- **Security considerations**
  - Resource served over HTTP now has access to GPU.
    - Limit loops
    - Non-normative validation guidance as part of specification
      - User Interface Questions

- **Some “in-built” Shaders for the fixed-function pipeline**

- **Much discussion in the WG about Portable Shaders**
  - Taking base from GLSL ES
  - Limiting/removing Features, subject to validation

Completing the Pipeline

- **What should tools (e.g. Maya) emit?**
  - JavaScript? Content formats?

- **Libraries on the Web**
  - We are starting with the low-level piece, looking forward to the evolution of libraries
  - Introducing capabilities, and watching benchmark numbers

- **Multiple interoperable implementations**
  - Demos exist