Slides, Filters and Mobile Oh My!
(Demonstrations of HTML and Declarative 3D Integration)

Sandy Ressler
sressler@nist.gov

National Institute of Standards and Technology

Gaithersburg MD 20899

ABSTRACT

This paper describes a set of demonstrations that illustrate some of the advantages of declarative 3D. Putting descriptions of 3D geometry directly into the DOM structure of a web document offers content creators the ability to use the rich infrastructure that already exists for the development and deployment of web pages. The three demonstrations illustrate the use of 3D in the context of a slide show (HTML as slide show substitute), of new CSS3 filtering capabilities, and finally the use of jQuery mobile to provide a user interface framework around pages containing 3D graphics.

Categories and Subject Descriptors

H.5.3 [**Information Interfaces and Presentation**]: Group and Organization Interfaces – *Web-based interaction.*

General Terms

Documentation, Design, Human Factors, Standardization, Languages.

Keywords

Web3D, declarative 3D, X3D.

1. INTRODUCTION

A collection of three demonstrations is described in this short paper. Each of the demos highlights an aspect of declarative 3D. Placing 3D geometry directly into the DOM of web pages allows content developers to leverage the rich infrastructure that already exists for the development and deployment of web pages. Each of these examples may be viewed live on the web (URLs are listed in the descriptions). The three demonstrations cover the usage of declarative 3D in the context of 1) a slide show, 2) integrated with CSS content filtering and 3) integrated with jQuery mobile, a JavaScript framework for mobile devices. These particular demonstrations use X3DOM[1], however, the principles are generalizable to any declarative 3D technology.

2. Deck.js

URL: math.nist.gov/~SRessler/deckjs/introduction/index2x3d.html

Deck.js [2] is a clever little JavaScript framework that enables users to create slideshows within the context of a web browser. Simply code up your slides and present your show via a set of

web pages. Upon discovering this suite, naturally the question was can Web3D graphics be embedded into those types of slide show pages? The short answer is yes and this demo illustrates the results of this test.
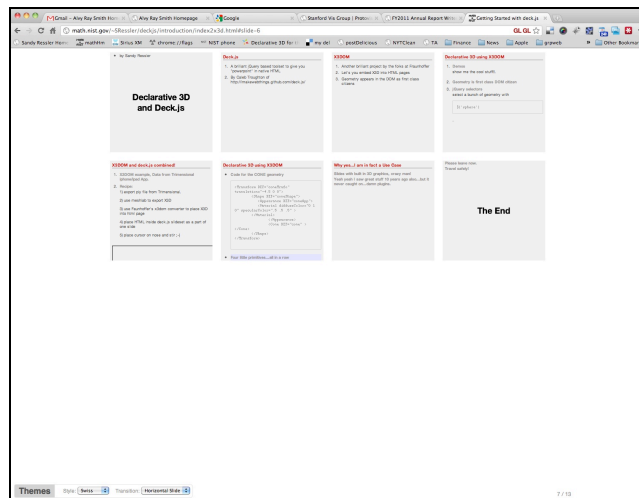

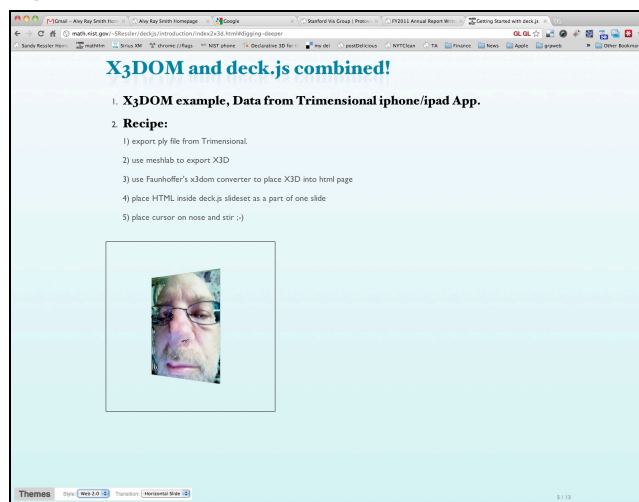**Figure 1. Overview of HTML "slideshow"**

Figure 2. Slide with embedded 3D content

3. CSS3 Content Filtering

URL: math.nist.gov/~SRessler/x3dom/CSS3-Content-Filter/index.html

The new version of CSS3 offers the ability to filter content based on the class attribute of an element. In this example, modeled after Gupta[3], the actual filtering is via "negation pseudo-class" combined with the class attitute. In code it looks like:

```
a[class="threed"]:focus ~ div:not( [class="threed"])
```

```
{
    height:0px; width:0px; border: none; margin:0;
}
```
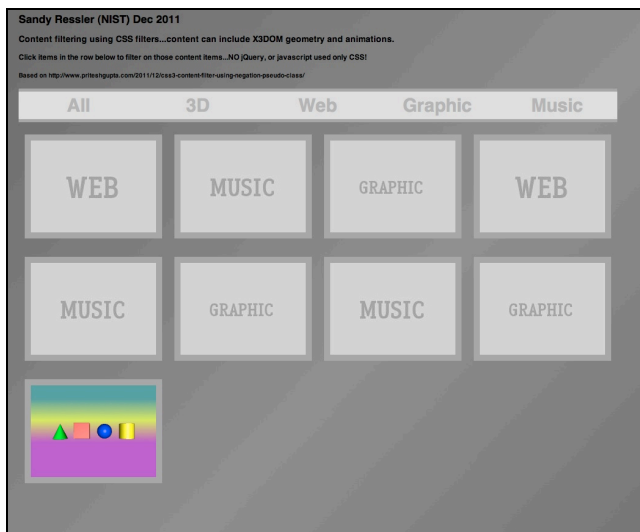

Figure 3. HTML page with content types including embedded 3D

Figure 4. jQuery mobile index page

4. jQuery mobile

URL: math.nist.gov/~SRessler/x3dom/jqmobile.html

jQuery mobile [4] is "A unified, HTML5-based user interface system for all popular mobile device platforms, built on the rock-solid jQuery and jQuery UI foundation." This demonstration was created to see if the jQuery mobile framework could in fact work with pages that have declarative 3D embedded in the page. As the demo illustrates the answer is yes, however, there were issues initially. On first loading the slideshow the 3D content displayed once and then disappeared. It turns out that, perhaps due to a peculiarity of X3DOM or the peculiarities of the browser (Chrome in this case), the graphics much be cached and remain in the cache. Fortunately jQuery mobile has a function to "pre-cache" page content and this solved the problem.
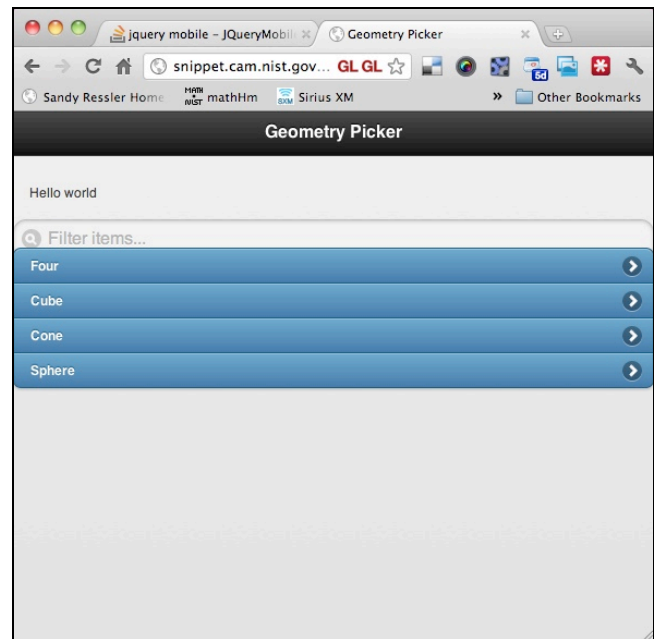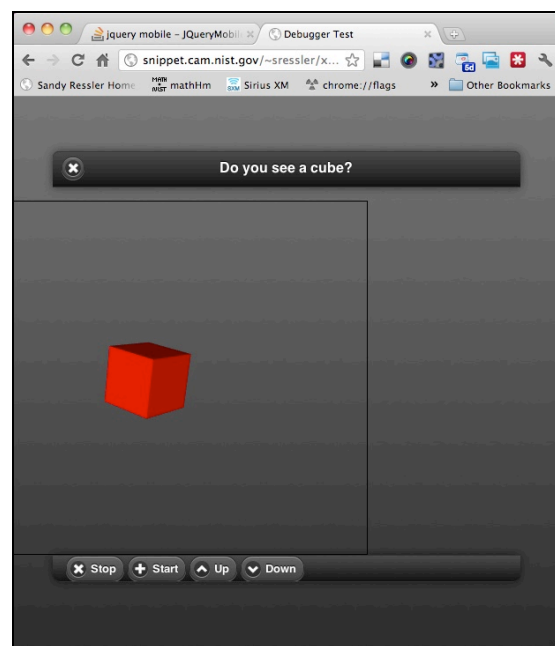

Figure 5. jQuery mobile 3D content page with active buttons

5. REFERENCES

[1] Johannes Behr , Peter Eschler , Yvonne Jung , Michael Zöllner, X3DOM: a DOM-based HTML5/X3D integration model, Proceedings of the 14th International Conference on 3D Web Technology, June 16-17, 2009, Darmstadt, Germany [doi>10.1145/1559764.1559784]

[2] Deck.js http://imakewebthings.github.com/deck.js/

[3] Gupta, http://www.priteshgupta.com/2011/12/css3-content-filter-using-negation-pseudo-class/

[4] jQuery mobile, http://jquerymobile.com