X3DOM DEMO

(pronounced X-Freedom [1])

WebGL Seminar, 11.03.2011
Tampere University of Technology

Juha Nurmi
Tampere University of Technology
Hypermedia Laboratory

Abstract

Standard web browsers can be used to serious software development [2]. This simple demonstration will show you that we can easily program interactive 3D web-based application. Demo program is done by integrating X3D nodes into HTML5 DOM content with X3DOM Javascript library [2].

Table of contents

- Introduction
- Experimental procedure
- Results
- Discussion
- Conclusion

Introduction

- This demo program is done by using X3DOM

  (WebGL seminar presentation - X3DOM. Friday, January 7, 2011. Jaakko Salonen [4])
- The X3DOM library is written in Javascript
- It is open source framework for WebGL
- X3D scene can be manipulated through HTML DOM
- 3D content is inside <x3d> tag

Experimental procedure

- Simple 3D worm game
- Tries to test most of the properties of X3DOM

Experimental procedure

Requirements:

- No special knowledge about 3D is needed
- Basic knowledge about HTML DOM
- Basic knowledge about Javascript
- The web browser that supports WebGL
- Vector javascript library ”sylvester.js”
- Keyboard javascript library ”shortcut.js”

Results

- working and playable worm game
- http://www.students.tut.fi/~nurmi28/wormgame/wormgame.html
- Whole process was quite straightforward
- Only 600 lines of written code

Results – how it works

- Creates 50 blue spheres at random positions with some transparency
- Worm is made of 50 red spheres
- Every 50ms interval the tail sphere is removed and a new head sphere is added
- X3D viewpoint is also changed and it follows the worm

Results – how it works

- Using vectors we can define the place of the sphere
- And give direction vector where to draw a new head
- Users keyboard input changes the direction vector

Results – how it works

- If the worms head is inside a blue sphere, the sphere disappears and score is added
- This is checked every 2000ms

Results – how it works

- Background images
- <group> element for the worm
- <group> element for the targets

Discussion

- There is no documentation for X3DOM, absolutely nothing!
- Fog does not work
- Only some attributes of X3D works
- Onmouseover did not work when I started to program this demo – it is now working
- 3D CSS support was also poor – works also better now

Discussion

- Using DOM inside <x3d> element does not always work right
- Double click does not work
- Onclick works right if you add

  <scene **pickMode="box"**>
- Onclick hits wrong objects sometimes

Discussion

- I tried to save the whole content of <scene> element to HTML5 local storage (save game) and wrote the content back to <scene> element (load game)

  => breaks the X3DOM library
- Although, saving only the <groub> element content and loading it back does work
- Inline X3D content is not dynamic

Conclusion

- Programming with X3DOM is like programming with Javascript and SVG
- It is quite easy to learn
- X3DOM is not ready DOM yet
- With properties of HTML5 it can be used to 3D application programming within the web browser

List of references

1. http://www.x3dom.org/?page_id=2
2. http://www.web3d.org/x3d/wiki/images/3/30/X3dom-web3d2009-paper.pdf
3. http://lively.cs.tut.fi/seminars/WebGL2011/WebGL-seminar-2010-1209.pdf
4. http://lively.cs.tut.fi/seminars/WebGL2011/x3dom-salonen.pdf

Appendices

- Worm game HTML document file:

  http://www.students.tut.fi/~nurmi28/wormgame/wormgame.html

- My Javascript code file:

  http://www.students.tut.fi/~nurmi28/wormgame/program_javascript/game.js