

WebGL 데모 및 효과 분석

- 김민태(케이티하이텔 iOS팀 및 Web플랫폼Lab)
- 이정표(케이티하이텔 플랫폼사업부)
- 윤용익(숙명여자대학교 멀티미디어학과)

I. 서론

WebGL은 OpenGL 표준을 담당하는 Khronos Group에서 담당하고 있는 표준 Web용 그래픽 라이브러리이다[1]. 최근 들어 관련 기술들이 상당히 발전해서, 웬만한 프로그램이나 게임은 다 만들 수 있는 수준이 되었다는 평가가 지배적이다. WebGL은 지금까지 조사해본 바로는 현재 크롬에서 가장 많이 지원하고 있으며 IE를 제외한 다른 모든 브라우저들이 지원하고 있다. WebGL이라는 이름에서 할 수 있듯이 내부적으로 OpenGL을 사용하고 java script를 이용해서 API가 제작되어 있다. 단순히 기본적인 렌더링만 되는게 아니라 texture는 물론, shader 프로그래밍까지 지원하고 있다

본 연구에서는 N-스크린 스마트 콘텐츠를 포함한 웹 애플리케이션을 제작함에 있어서 WebGL기반의 3D Graphics 구현에 대한 전반적인 현황을 여러 가지 데모를 통하여 분석한 결과를 설명한다.

II. 웹 브라우저 동향

최근 다양한 스마트 기기들의 등장과 함께 다양한 웹브라우저 사용이 늘어나면서 IE 하락 추세 속에 세계 웹브라우저는 3강구도로 재편되고 있다.

2011년 12월 웹 분석업체인 스탯카운터 통계에 따르면 IE의 세계 시장 점유율은 38.61%로 30%대를 기록했고 크롬(27.2%)과 파이어폭스(25.4%)가 그 뒤를 이었다. 이는 2010

년 12월 IE가 46.9%, 파이어폭스와 크롬이 각각 30.7%, 14.8%를 차지했던 것과 비교해 크게 달라진 것이다. 특히 크롬은 1년새 10%가 넘게 성장, 지난해 들어 파이어폭스까지 앞질렀다. 우리나라는 여전히 IE가 압도적인 1위를 차지하고 있다. 24일 기준으로 한국 웹브라우저 점유율은 IE가 81.02%이며, 그 뒤를 크롬(11.36%), 파이어폭스(4.8%)가 잇고 있다. 하지만 전체적인 추세는 IE의 점유율이 줄어드는 양상이다.

스탯카운터 집계에 따르면, 올 초 93%에 육박하던 IE의 점유율은 9월 들어 89.6%로 내려왔고 이 달 24일 기준으로 81.2%까지 하락했다. 반면, 올 초 3%대(3.18%)에 불과했던 크롬이 11.3%로 10%대를 넘는 등 해외뿐 아니라 국내에서도 선전하는 모습을 보여줬다.

지난 2010년 1월 97%의 점유율을 기록했던 IE가 점차 하강세를 유지하고 있다는 점과 크롬의 선전은, 점차 국내 웹브라우저 시장도 탈IE 분위기로 흐를 수 있다는 점을 보여주고 있다. 또 그동안 국내는 보안 때문에 IE를 기반으로 인터넷뱅킹이 활성화됐지만, 최근 다양한 스마트 기기들의 등장과 맞물려 오픈뱅킹, 다양한 웹브라우저 사용이 늘어나면서 이 같은 IE하락 추세는 이어질 전망이다[2].

III. WebGL on Internet Explorer

1. IEWebGL

세계 시장 점유율 1위이자 국내 및 중국 등 아시아 시장에

서 독보적인 점유율을 갖고 있는 마이크로소프트의 Internet Explorer가 내부적으로 WebGL 지원을 위한 작업에 착수, 2011년 6월 27일 기존 IE에 플러그인 방식의 IEWebGL 베타버전을 공개하였으며 2012년 5월 이 버전은 IEWebGL v1.0.8 Released까지 발전하였다[3].

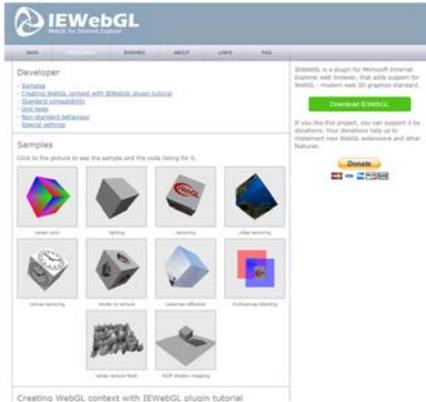


그림 1. IEWebGL

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
functionInitWebGLContext() {
    varglCanvas = document.getElementById("glCanvas");
    // first, try standard WebGL context
    vargl = glCanvas.getContext("webgl");
    if(!gl) {
        // if failed, try experimental one
        gl = glCanvas.getContext("experimental-webgl");
    }
    if(!gl) {
        alert("Your browser does not support WebGL");
        return;
    }
    // here we get WebGL context -
    // for demonstation let's show some info
    alert(
        "WebGL version="+ gl.getParameter(gl.VERSION)
        + "\n"+
        "WebGL vendor="+ gl.getParameter(gl.VENDOR)
        + "\n"+
        "WebGL renderer="+ gl.getParameter(gl.RENDERER)
        + "\n"
    );
}
</script>
</head>
<bodyonload="InitWebGLContext()">
<objectstyle="width:100%;height:100%"id="glCanvas"
type="application/x-webgl"></object>
</body>
</html>
```

그림 2. WebGL context 생성 방법

3.1 WebGL context 생성 방법

IE에 IEWebGL 인스톨 후, 그림 2와 같이 Canvas element 대신 Object element를 생성하여 구현한다.

3.2 Inlining into markup

현재 브라우저가 WebGL을 지원하는지를 감지하여, 지원 하지 않거나 브라우저가 IE일 경우, 그림 3과 같이 Object element를 생성하고 IEWebGL 플러그인을 로드한다.

```
<!DOCTYPE html>
<html>
<head>
<script type="text/javascript">
functionInitWebGLContext() {
    varglCanvas = document.getElementById("glCanvas");
    // first, try standard WebGL context
    vargl = glCanvas.getContext("webgl");
    if(!gl) {
        // if failed, try experimental one
        gl = glCanvas.getContext("experimental-webgl");
    }
    if(!gl) {
        alert("Your browser does not support WebGL");
        return;
    }
    // here we get WebGL context -
    // for demonstation let's show some info
    alert(
        "WebGL version="+ gl.getParameter(gl.VERSION) +
        "\n"+
        "WebGL vendor="+ gl.getParameter(gl.VENDOR) +
        "\n"+
        "WebGL renderer="+ gl.getParameter(gl.RENDERER)
        + "\n"
    );
}
</script>
</head>
<bodyonload="InitWebGLContext()">
<canvasstyle="width:100%;height:100%"id="glCanvas"></
canvas>
</body>
</html>
```

그림 3. Inlining into markup

2. 호환성

IEWebGL은 WebGL v1.0 스펙을 완벽히 소화하며 그림 4, 그림 5와 같이 JavaScript debugger를 지원한다.

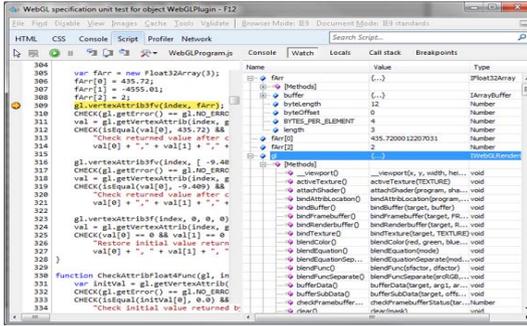


그림 4. JavaScript debugger



그림 6. WebGL 3D Demo 화면(IE9)

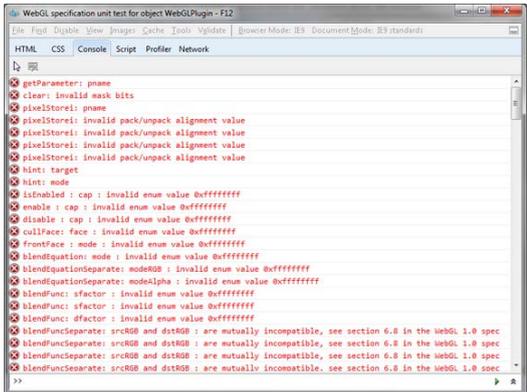


그림 5. Console error logging

IV. WebGL on Chrome (Google)

3D 그래픽은 ActiveX도 플래시도 아닌 웹브라우저 자체에서 구동되는 WebGL기술로 구현된 데모 화면으로 그림 7 과 같이 고해상도를 제공한다.



그림 7. 고해상도 3D 화면

그림 8과 같은 WebGL을 이용한 3D 게임이 지속적으로 개발 중이기 때문에 컴퓨터 성능, 브라우저 성능 이외의 몇 가지 요소들만 갖추어진다면 다양한 게임 분야 및 그래픽 분야 등에 활용도가 높을 것으로 보인다.

3. Non-standard behaviour

3.1 Texturing

IE6, IE7과 IE8 브라우저는 HTML5 동영상과 canvas를 지원하지 않는다. 따라서 해당 프로젝트로부터 Texture는 생성될 수 없다.

3.2 Events

IE6, IE7, IE8 WebGL의 webglcontext creationerror, webglcontextlost와 webglcontext restored event들은 onerror event를 사용하여 구동하게 된다.

4. DEMO on IEWebGL (plungin, IE9)

IE9에서 실행되는 WebGL 3D Demo 결과는 그림 6과 같다.



그림 8. Cloudmach사의 3D 소셜 게임 화면[4]

WebGL Terrain Editor[5]로 지형을 제작한 결과는 그림 9와 같다.

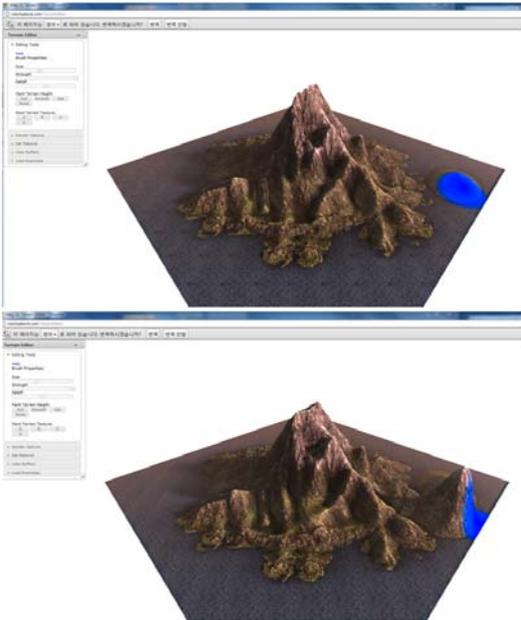


그림 9. WebGL Terrain Editor로 제작한 지형 화면

오른쪽 파란색 영역으로 지형을 생산하는 시뮬레이션 결과를 볼 수 있다.

1. WebGL Shader Demo : 물결

Shader를 이용하여 I-시스템, 프랙탈 끌어당기기와 같은 데모들을 생성한 결과는 그림 10이다. 그래픽 카드에서 실행되는 WebGL 픽셀 셰이더 언어인 GLSL을 이용하여 생성한 것이다. 이 방법은 화면에 출력되는 비트맵의 모든 픽셀을 매우 빠르게 처리할 수 있다.

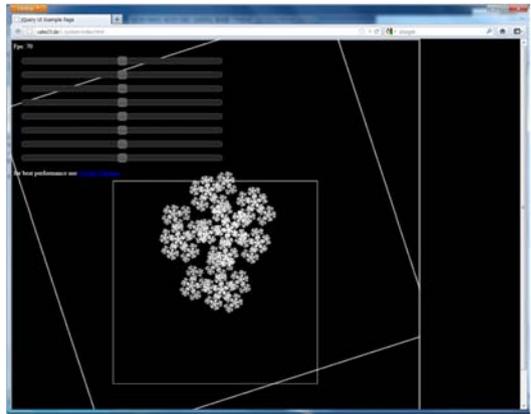


그림 10. WebGL Shader 구현 화면

그림 11은 Fine-tuned 8Bit Reaction-Diffusion 시스템에 순회하는 물결들과 서브픽셀 붕괴(subpixel decay)효과를 추가하고, 광원의 범위 매핑을 위해서 그라디언트로 생성한 텍스처를 사용한 결과이다.

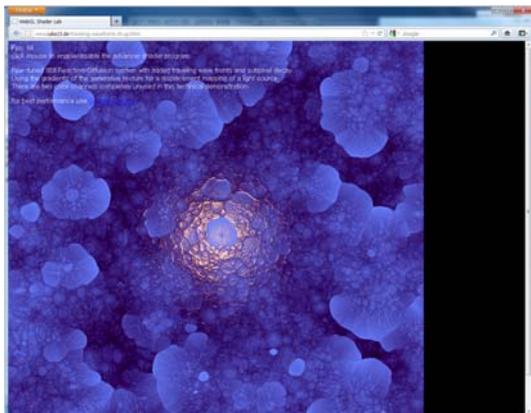


그림 11. 순회하는 물결과 서브픽셀 붕괴 효과 추가 화면

2. WebGL Shader Demo : 구름

three.js라이브러리를 이용하여 Shader를 처리한 결과는 그림 12이다.



그림 12. three.js라이브러리를 이용한 Shader 처리 화면

그림 12는 파이어폭스4, 사파리10.6, 구글 크롬10 이상에서 정상적으로 실행한다. 마우스를 포인트하는 방향으로 전체 화면이 이동을 합니다.

V. WebGL on Firefox (Mozilla)

Mozilla Firefox 브라우저의 최신 버전은 PC 데스크톱 버전 및 모바일에서 부가 기능을 동기화 할 수 있다. 또한 웹 페이지 요소 시각화 및 CSS를 실시간으로 편집이 가능한 새로운 웹 개발 도구를 통합하였다.

Firefox는 Firefox Sync를 확장하고 부가 기능 동기화 기능을 구현했다. 따라서 여러 컴퓨터에서 부가 기능을 동기화 하면 회사와 집에서 Firefox에서 동일한 브라우징 환경을 간단히 만들 수 있다. 이 기능은 그림 13 설정 화면 Sync 탭에서 활성화할 수 있다.

초기 단계에서는 동기화할 수 있는 기능을 제한하고 다음 조건을 모두 만족하는 것만 동기화 하도록한다.

- 기능 유형이 확장 기능이나 테마인 경우
- Mozilla 공식 기능 소개 사이트에서 설치된 것
- 기능 소개 사이트 상태가 정식 공개된 것
- 사용자 본인이 자신의 프로필에 설치한 것



그림 13. 동기화 설정 화면

웹 개발자를 위한 주요 기능은 Firefox 자체의 강력한 시각화 도구로 웹 페이지 속성에 3D 보기를 추가한 것이다. 이것은 WebGL을 활용하여 완전히 새로운 웹 페이지 시각화 도구로서 페이지의 구조를 평면이 아닌 입체적으로 표현한다. 그림 14는 웹 페이지 속성에 3D 보기를 설정한 것이다.

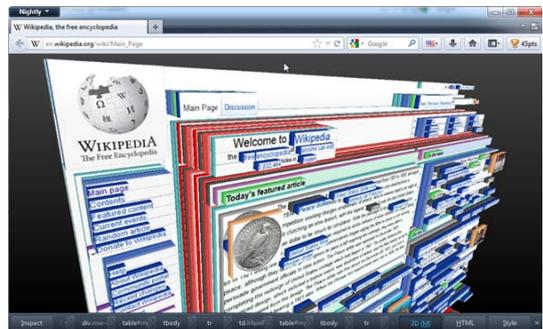


그림 14. 3D 보기 화면

3D 보기 기능을 소스 코드와 실제 레이아웃 관계를 즉각적으로 파악할 수 있습니다. 소스와 같은 기존 도구 페이지를 만드는 방법을 배우고 싶을 때 언제든지 3D 보기에서 페이지의 각 부분이 어떤 구조로 되어 있는지를 분명하게 알 수 있다. 페이지 관리자틀 열고 3D를 선택한 후, 요소에 마우스 오버하면 해당 부분에 대한 자세한 정보를 확인할 수 있습니다.

1. Firefox WebGL Demo

그림 15는 Nokia N900에서 WebGL 시물레이션결과 화면이다.



그림 15. WebGL 시물레이션 화면(Nokia N900)

그림 16은 Windows Phone 7에서 WebGL Game시물레이션 결과 화면이다.



그림 16. WebGL Game 시물레이션 화면(WindowsPhone 7)

그림 17은 WebGL 증강현실 시물레이션 결과 화면이다.



그림 17. WebGL 증강현실 시물레이션 화면

VI. 결론

본 연구에서는 표준 Web용 그래픽 라이브러리인 WebGL 기반의 3D 그래픽 구현 현황을 다양한 시물레이션 결과를 통하여 분석하였다.

IE6, IE7과 IE8 브라우저는 HTML5 동영상과 canvas를 지원하지 않지만 IE9에서는 WebGL 3D 를 지원한다. 그리고 크롬 브라우저는 3D 그래픽을 웹브라우저 자체에서 구동되는 WebGL기술로 구현하였다. Firefox 브라우저는 PC 데스크톱 버전 및 모바일에서 부가 기능을 동기화 할 수 있으며, 웹 페이지 요소 시각화 및 CSS를 실시간으로 편집이 가능한 새로운 웹 개발 도구를 통합하였다.

본 연구 결과는 N-스크린 스마트 콘텐츠를 포함한 웹 어플리케이션을 제작함에 있어서 WebGL 기반의 3D 그래픽 구현에 기초적인 자료가 될 것이다.

참고문헌

- [1] <http://www.khronos.org/webgl/>
- [2] 디지털타임스 http://www.dt.co.kr/contents.html?article_no=2011122702011160746005
- [3] <http://iewebgl.com/>
- [4] <http://cloudmach.com/>
- [5] <http://www.earthsculptor.com/>

※ 이 논문은 한국콘텐츠진흥원의 “2011년 콘텐츠산업 기술지원사업”의 지원으로 연구된 결과입니다.

저 자 소 개



김민태

1991: 정석항공공업고등학교
항공 전자과.
1995~1997: 한국신용평가
1997~1999: 보엔엘소프트
개발 팀장
1999~2000: 한국기술표준원
연구원
2000~2004: (주)투비웨이
개발 선임
2004~2011: (주)이미지클릭
개발 팀장
현 재: 케이티하이텔 개발실
iOS팀 및 Web플랫폼
Lab 팀장
W3C HTML5 KIG UI
그룹장
관심분야: Web & 모바일 플랫폼,
모바일컴퓨팅,
클라우드컴퓨팅,
N-Screen 서비스



이정표

1998: 인하대학교
물리학과 이학사.
2002-2008: Teleca Korea
수석연구원.
2010: Teleca Korea 기술이사
현 재: 케이티하이텔(주)
플랫폼사업부 부장
관심분야: Web & 모바일 플랫폼,
모바일컴퓨팅,
클라우드컴퓨팅,
N-Screen 서비스



윤용익

1983: 동국대학교
통계학과 이학사.
1985: 한국과학기술원
전산공학과 공학석사.
1994: 한한국과학기술원
전산공학과 공학박사
현 재: 숙명여자대학교
멀티미디어학과 교수
관심분야: 스마트사아니지,
스마트 클라우드 컴퓨팅,
모바일 멀티미디어 시스템,
분산시스템,
실시간 처리시스템,
미들웨어,
실시간 OS/DBMS,
상황인지 서비스,
N-Screen 표준화,
모바일 클라우드