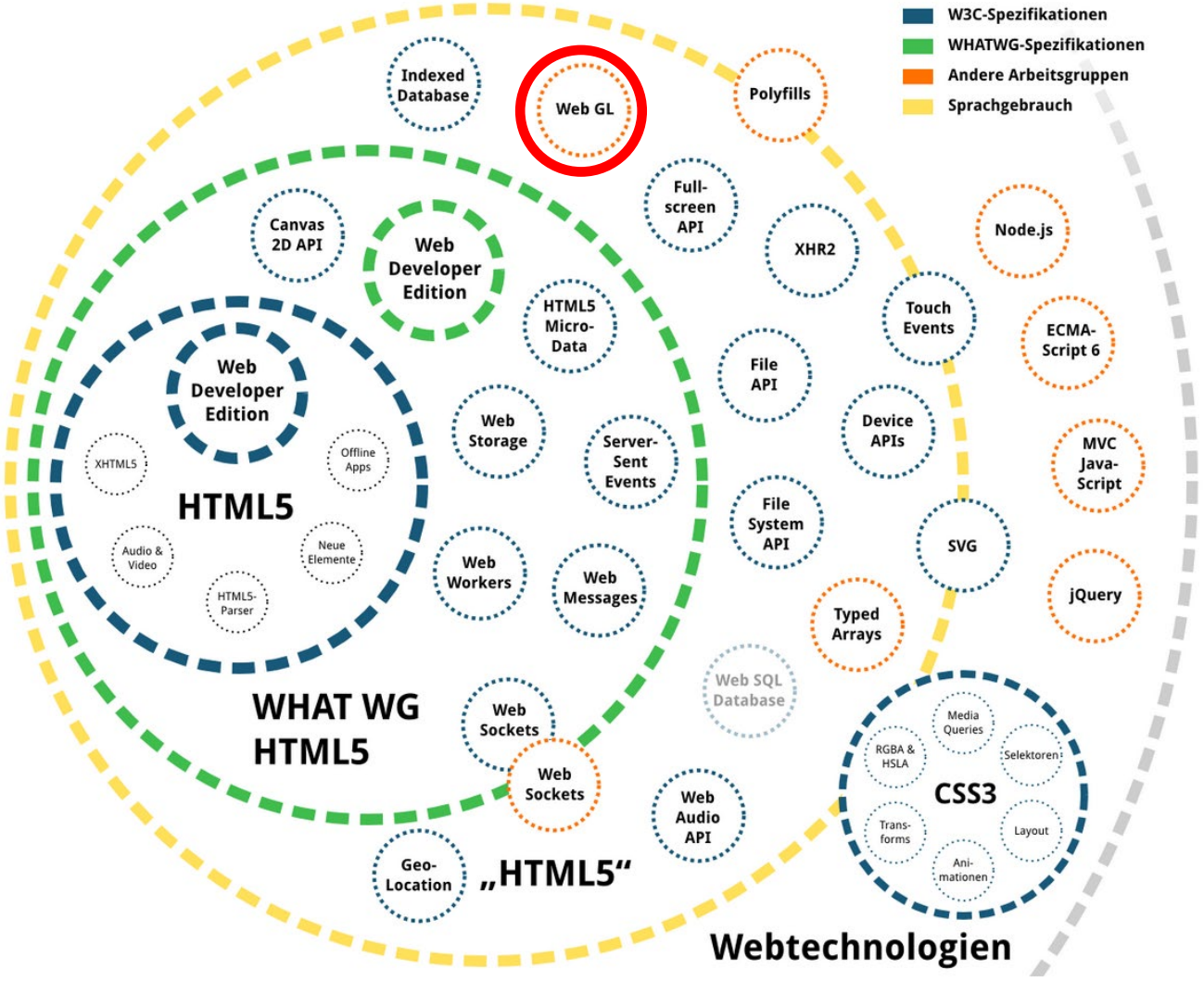
WebGL: 3D-Modelle im Browser nativ visualisieren

Prof. Dipl.-Ing. Martin Schober, Hochschule Karlsruhe – Technik und Wirtschaft

Webtechnologien um HTML5



Quelle: Peter Kröner, http://html5-buch.de

WebGL – Das Format

Mit der Technologie **WebGL** besteht die Möglichkeit, nativ, also ohne PlugIn, die dritte Dimension im Browser mit **JavaScript und OpenGL** zu visualisieren.

Die wichtigsten Browser-Anbieter wie Apple (Safari), Google (Chrome), Mozilla (Firefox) und Opera (Opera) sind Mitglieder der **WebGL-Arbeitsgruppe**.

Die **Khronos Group**, ein Non-Profit-Industriekonsortium, ist Mitglied der WebGL-Arbeitsgruppe und entwickelt den Standard.

Khronos Group → AMD, Apple, ARM, Ericsson, Intel, Nokia, NVIDIA, Samsung und Sony

WebGL - Hintergrund

3D Technologie zur Anzeige von **hardwarebeschleunigten** 3D-Graphiken.

Hardwarebeschleunigt heisst, der Browser greift direkt auf die Rechenleistung der Graphikkarte zu, um komplexe 3D-Inhalte anzuzeigen.

Auf der Code-Ebene funktioniert das mit **HTML5** Elementen. Es wird **JavaScript** benutzt um auf Funktionen einer angepassten Version von OpenGL ES 2.0 (Open Graphics Library for Embedded Systems) zuzugreifen.

WebGL soll 3D-Daten langfristig vor allem auf mobile Endgeräte bringen!

Browserunterstützung

Folgende Browser unterstützen WebGL:

- Google Chrome unter Windows
- Firefox unter Windows und Linux
- Opera 12.12 unter Windows
- Internet Explorer 10 unter Windows
- Safari unter MAC OS
- Opera Mobile unter Android
- Firefox unter Android

Weitere Hilfe zu Browsereinstellungen:
http://webgl-publisher.com/BrowserConfigDe.html

WebGL Grundaufbau

WebGL ist kein Teil von HTML5, jedoch erfolgt der Zugriff auf die WebGL API über einen speziellen WebGL-Context eines Canvas Elements:

```
canvas.getContext("webgl").
```

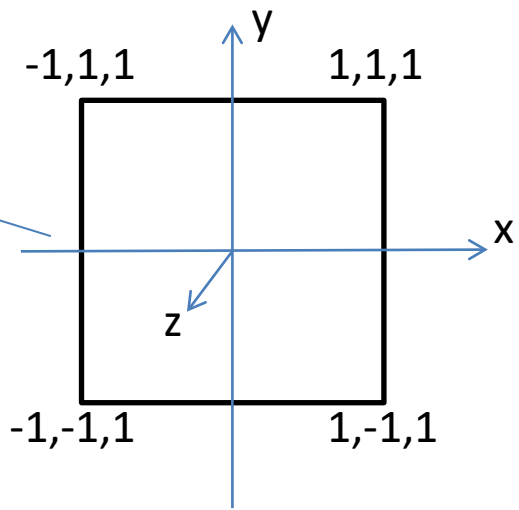
Grafiken, welche mittels der WebGL API gezeichnet werden, werden also ganz konventionell über ein Canvas Element dargestellt.

WebGL Grundaufbau - Würfel

Quelle: https://developer.mozilla.org/de/docs/WebGL/

Die Vertex-Positionen des Würfels definieren

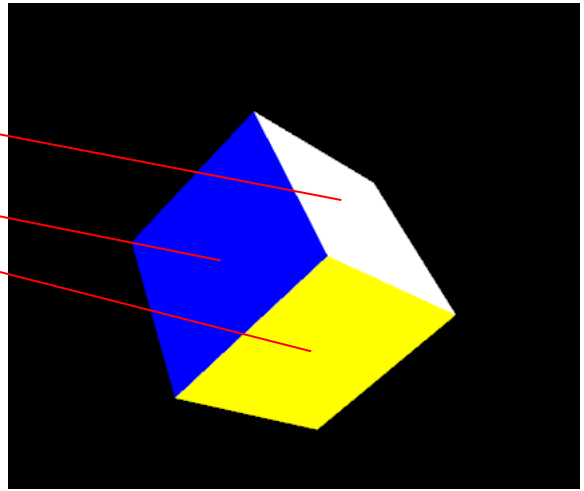
```
var vertices = [
    // vordere Fläche
    -1.0, -1.0,  1.0,
     1.0, -1.0,  1.0,
     1.0,  1.0,  1.0,
    -1.0,  1.0,  1.0,

    // hintere Fläche
    -1.0, -1.0, -1.0,
    -1.0,  1.0, -1.0,
     1.0,  1.0, -1.0,
     1.0, -1.0, -1.0,

    // obere Fläche
    -1.0,  1.0, -1.0,
    -1.0,  1.0,  1.0,
     1.0,  1.0,  1.0,
     1.0,  1.0, -1.0,

    // untere Fläche
    -1.0, -1.0, -1.0,
     1.0, -1.0, -1.0,
     1.0, -1.0,  1.0,
    -1.0, -1.0,  1.0,

    // rechte Fläche
     1.0, -1.0, -1.0,
     1.0,  1.0, -1.0,
     1.0,  1.0,  1.0,
     1.0, -1.0,  1.0,
```



WebGL - Grundaufbau

Die Farben der Vertices definieren

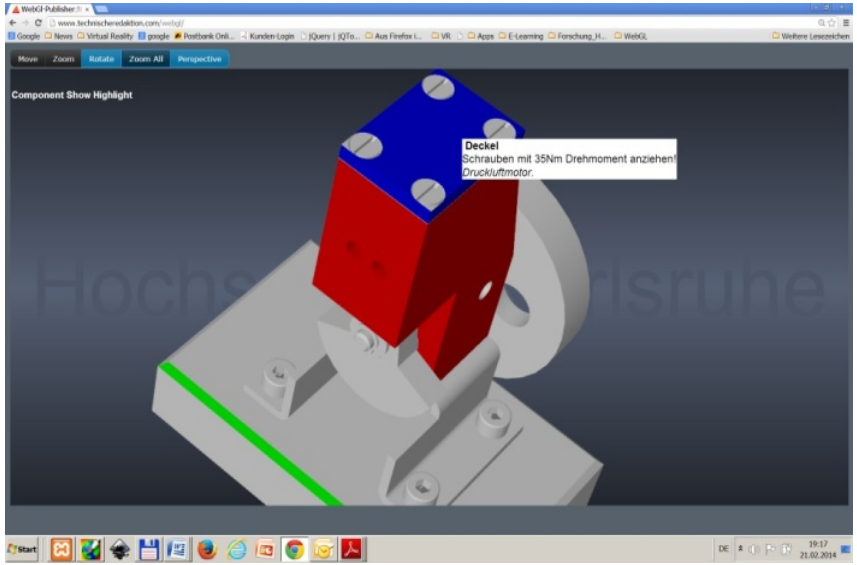
```
var colors = [
    [1.0,  1.0,  1.0,  1.0],    // vordere Fläche: weiß
    [1.0,  0.0,  0.0,  1.0],    // hintere Fläche: rot
    [0.0,  1.0,  0.0,  1.0],    // obere Fläche: grün
    [0.0,  0.0,  1.0,  1.0],    // untere Fläche: blau
    [1.0,  1.0,  0.0,  1.0],    // rechte Fläche: gelb
    [1.0,  0.0,  1.0,  1.0]     // linke Fläche: violett
  ];

  var generatedColors = [];

  for (j=0; j<6; j++) {
    var c = colors[j];

    for (var i=0; i<4; i++) {
      generatedColors = generatedColors.concat(c);
    }
  }

  cubeVerticesColorBuffer = gl.createBuffer();
  gl.bindBuffer(gl.ARRAY_BUFFER, cubeVerticesColorBuffer);
  gl.bufferData(gl.ARRAY_BUFFER, new WebGLFloatArray(generatedColors), gl.STATIC_DRAW);
```



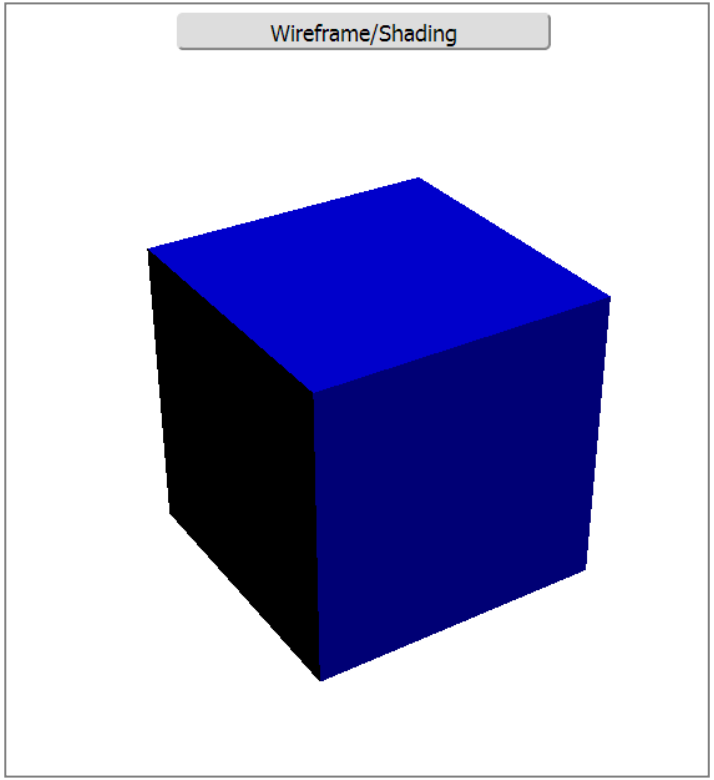
Erstellen von WebGL-Modellen

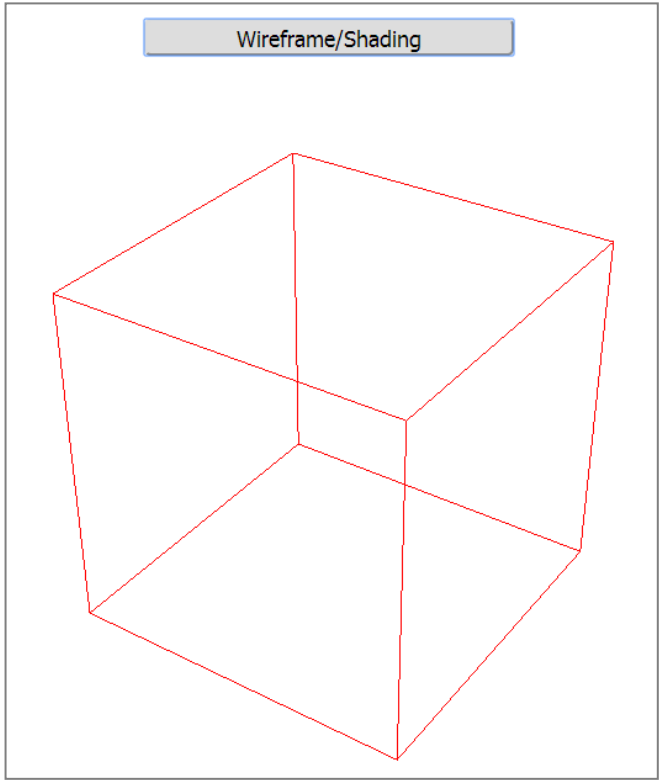
1. WebGL-Frameworks wie z.B. Three.js, SceneJS, PhiloGL und weitere, sowie

2. die Generierung von WebGL aus CAD-Werkzeugen wie Blender, WebGL-Publisher oder wie dem Programm Revit® von Autodesk® mit dem AddIn WebGL Publisher Export.



WebGL – Einfaches Beispiel mit Three.js





WebGL – Einfaches Beispiel mit Three.js

```
// Funktion für die Größenänderung des Würfels in X-,Y- und Z-Richtung
      function cubeScale()
            {
            cube.scale.x *= 1.2;
            cube.scale.y *= 1.2;
            cube.scale.z *= 1.2;
            }

// Überprüfung der Mausklicks auf die Webseite und Aufruf der cubeScale-Funktion zum Vergrößern
      document.addEventListener( 'click',function(){cubeScale()} , false );

// Funktion für die Änderung der Darstelung vom Flächenmodell zum Kantenmodell
       function change() {
            scene.remove(cube);

// Siehe nächste Folie
```

WebGL – Einfaches Beispiel mit Three.js

```
if (whichCube == 0 ) {
    cube = new THREE.Mesh(new THREE.CubeGeometry(100, 100, 100), new
    THREE.MeshBasicMaterial({
            wireframe: true,
            color: 'red'
           }));
           whichCube=1;
         }
else {
    cube = new THREE.Mesh(new THREE.CubeGeometry(100, 100, 100), new
    THREE.MeshLambertMaterial({
              shading: true,
              color: 'blue'
             }));
```

WebGL-Publisher (http://webgl-publisher.com/)

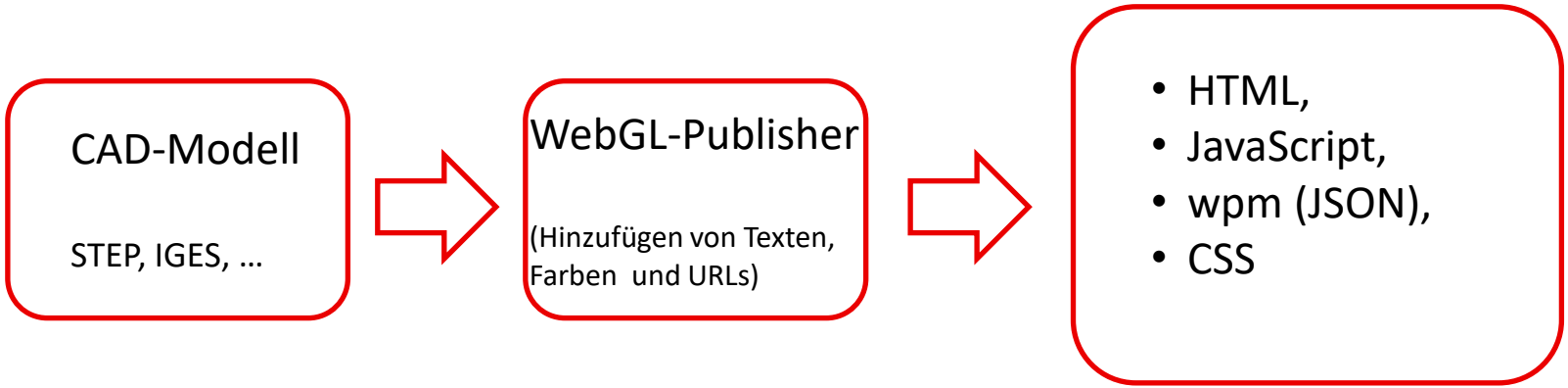
- WebGL-Publisher wird verwendet um Geometriedaten in einer 3D Form im Web zu publizieren.
- Die Software verwendet den WebGL Standard
- WebGL-Publisher kann verschiedene Geometrieformate importieren.
- Die importierte Geometrie kann in ihrer Darstellung durch Setzen von Farbe, Transparenz, Textureierung oder vordefinierten Shadern verändert werden.
- Die Geometrie kann in eine html Repräsentation exportiert werden.
- WebGL-Publisher kann unter Windows XP, Windows 7 bis Windows 10 betrieben werden und steht nur online als Download zur Verfügung.

WebGL-Publisher Importformate

Der Import eines 3D-Modells in das Tool WebGL-Publisher erfolgt über den Button *Öffnen in der oberen Menüleiste des Tools. Hier können 3D-Modelle folgender Formate importiert und danach bearbeitet werden (http://www.webgl-publisher.com/TechInfoDe.html):*

- DXF Geometrie (Linien, Kreise)
- Step
- IGES
- Wavefront (.obj)
- 3D Studio (.3ds)
- STL (.stl)
- CADMAI CAD Modelle (.cmi)

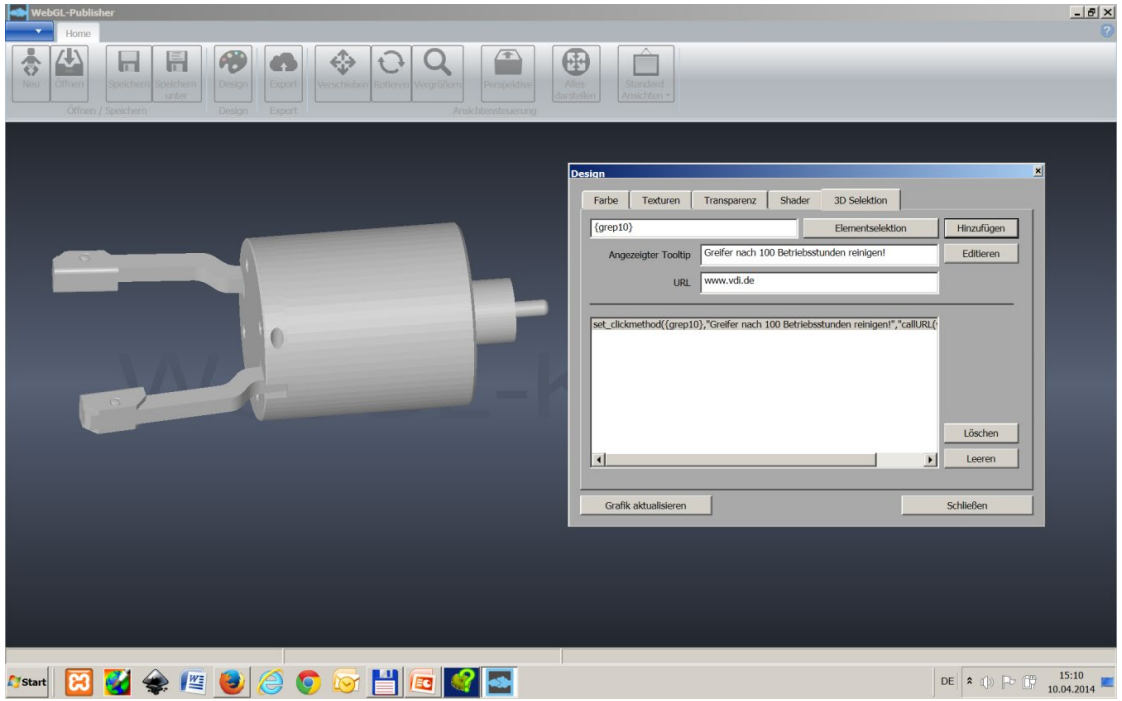
Vom CAD-Modell in das WebGL-Format



WebGL-Publisher

Einfügen von Erläuterungstexten und URLs

1. Modell importieren
2. Design-Werkzeug
   → Farben von Bauteilen ändern
3. Design-Werkzeug
   → Texte Bauteilen zuordnen
4. Design-Werkzeug
   → Bauteile mit Web-Seiten verlinken
5. Modell als WebGL-Datei exportieren

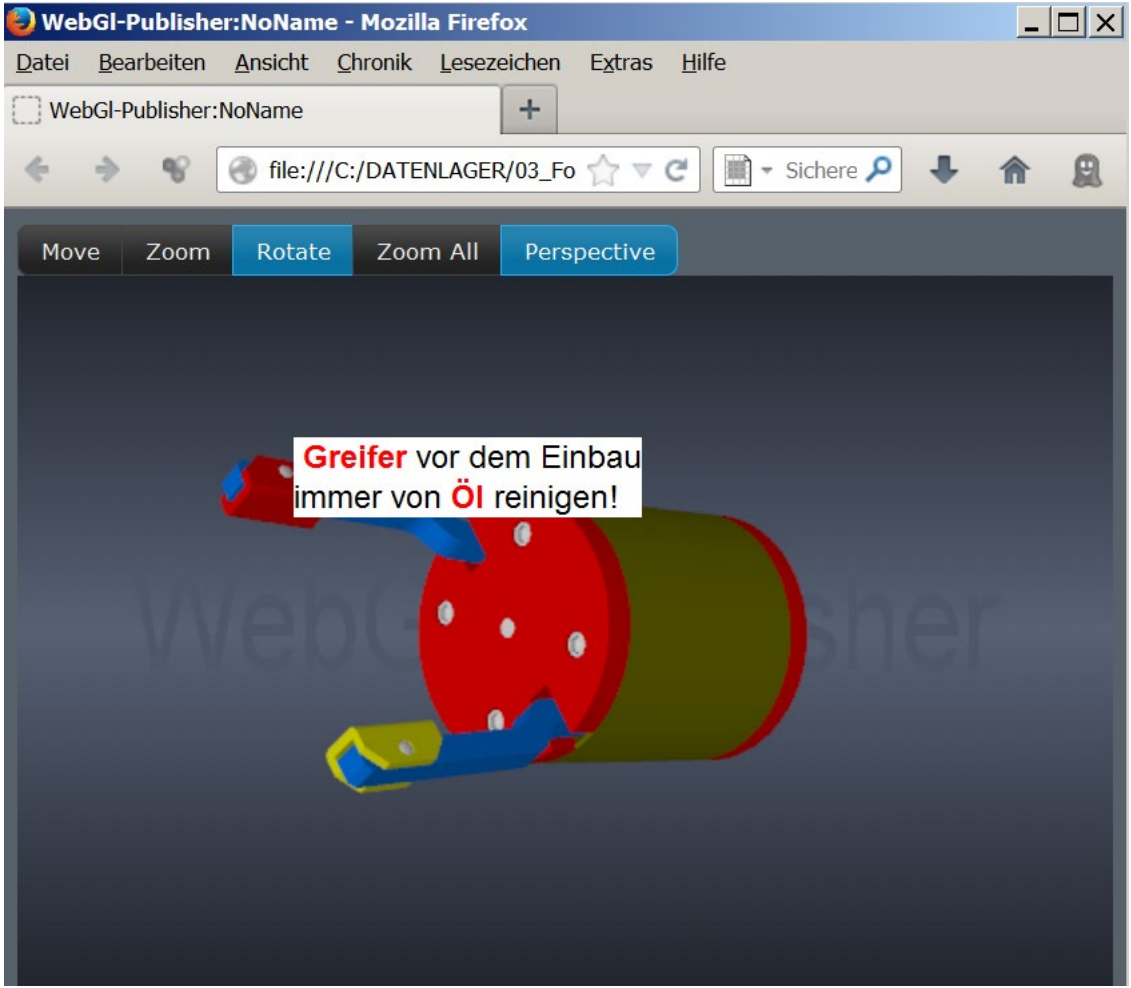


WebGL-Publisher

Modell im Browser testen:

Firefox ohne Webserver

Chrome mit Webserver (Apache)



WebGl-Publisher

Änderungen in der JSON-Datei vornehmen:

```
…
{\"clNm\":\"gR\",\
\"aNr\":1,\
\"eNm\":\"grep11\",\
\"sTx\":\"<b style='color:#F00'>Greifer</b> vor dem Einbau <br />immer von
   <b style='color:#F00'>Öl</b> reinigen!\",\
\"hRf\":\"http://www.tekom.de\",\
\"col\":[1.0,1.0,0.0,1.0],\
\"siZ\":0.486,\
\"cP\":[1.019,0.607,0.0],\
\"shNm\":\"Color3D\",\
...
```

HTML und CSS innerhalb der Geometrie!

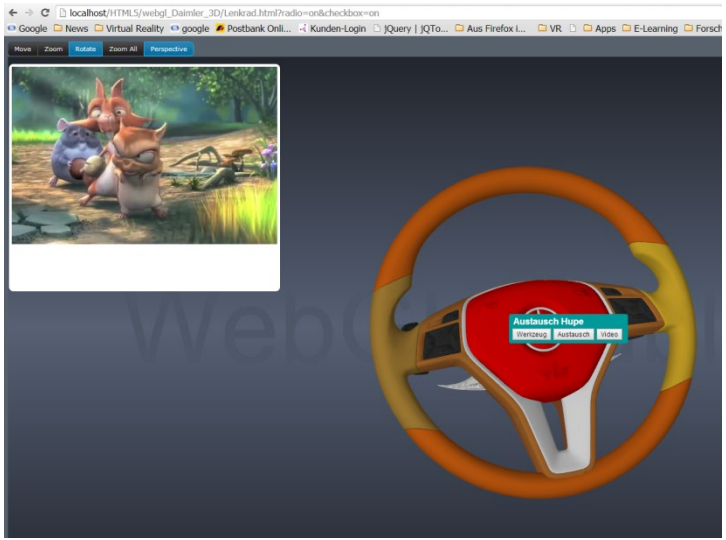
URL-Link innerhalb der Geometrie!

WebGl-Publisher – JavaScript-Funktionen integrieren

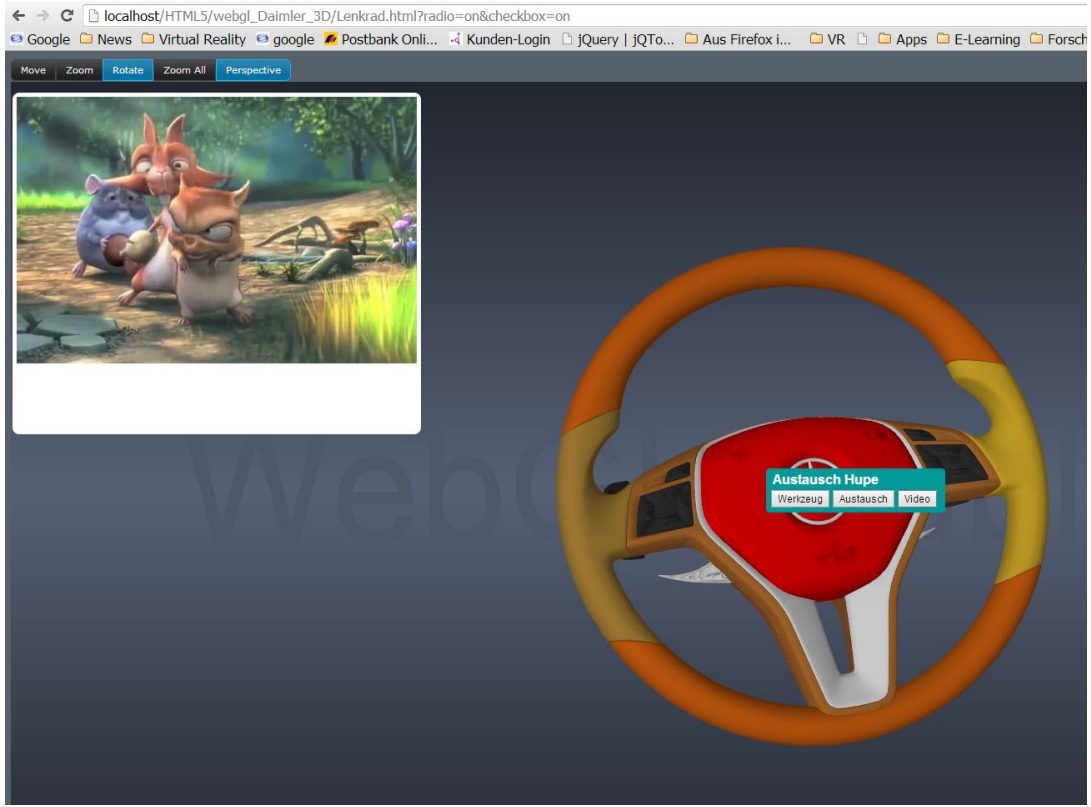
```
\"eNm\":\"grep33\",\
\"sTx\":\"<b>Austausch Hupe</b><br /><button id='allgemeinhupe'
    class='button' onmouseover='test1(1)'>Werkzeug</button><button
    class='button'  onmouseover='test1(2)'>Austausch</button><button
    class='button' onmouseover='test1(3)'>Video</button>\",\
\"col\":[0.78,0.48,0.2,1.0],\
\"siZ\":244.68,\
\"shNm\":\"Color3D\",\
```

Ereignisabfrage!

JavaScript-Funktion



WebGl-Publisher – JavaScript-Funktionen integrieren



Literatur und Links

Blender – Ihr Einstieg in die professionelle 3D-Grafik und Animation, Heiko Ihde Addison-Wesley Verlag
- http://www.blender.org/about/, 12.03.2014
- http://t3n.de/news/3d-inhalt-html-syntax-erzeugen-x3dom-webgl-344872/, 12.03.2014
- http://www.igd.fraunhofer.de/Institut/Abteilungen/Visual-Computing-System-Technologies/Projekte/X3DOM, 12.03.2014
- http://www.leed.ch/history/x3d/werkzeuge.htm, 12.03.2014
- http://www.leed.ch/history/x3d/zweck.htm, 12.03.2014
- http://www.web3d.org/realtime-3d/x3d/what-x3d, 10.03.2014
- http://x3dom.org/docs/dev/tutorial/blender_export.html, 4.3.2014
- http://x3dom.org/docs/dev/tutorial/dataconversion.html#dataconversion, 4.3.2014
- http://www.x3dom.org/?page_id=2, 4.3.2014
- http://blender.freemovies.co.uk/displaying-3d-models-in-web-pages-using-webgl/, 4.3.2014
- http://www.semayjohnston.com/example.html, 13.03.2014
- http://blog.mayflower.de/4585-Eine-Einfuehrung-in-ThreeJS-WebGL.html, 13.03.2014
- http://blog.teamtreehouse.com/the-beginners-guide-to-three-js, 13.03.2014
- http://aerotwist.com/tutorials/getting-started-with-three-js/, 13.03.2014
- http://ffwd.typepad.com/blog/2011/04/webgl-what-flavor-is-your-engine.html, 12.03.2014
- http://webglframeworks.org/framework-documentation/framework-list/, 13.03.2014
- http://www.khronos.org/webgl/, 13.3.2014
- http://www.khronos.org/webgl/wiki/Getting_Started, 13.03.2014
- http://caniuse.com/webgl, 13.03.2014
- http://www.khronos.org/webgl/wiki/Getting_a_WebGL_Implementation, 13.03.2014
- http://en.wikipedia.org/wiki/Threejs, 12.03.2014
- http://dev.opera.com/articles/view/porting-3d-graphics-to-the-web-webgl-intro-part-2/, 4.3.2014
- http://www.3d-ring.de/3d/tutorial.php?tutorial=25331cd9d6c, 4.3.2014
- http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.208.575&rep=rep1&type=pdf, 13.03.2014
- http://mandemeskel.wordpress.com/2013/08/19/mouse-events-raycasting-with-three-js/, 13.03.2014

Literatur und Links

Hauser, T./ Wenz, C. / Maurice, F. (2008): Das Website Handbuch. Markt + Technik Verlag, München, 248-349

Hogan, Brian P. (2011) HTML5 & CSS3. Webentwicklungen mit den Standards von morgen. 1. Aufl. Köln, O'Reilly Verlag GmbH & Co.KG.

iX Kompakt Webdesign (2012) HTML5 und CSS3. Hannover, Verlag Heise

Koch, Daniel. (2011) HTML5 - Grundlagen & Praxislösungen. Düsseldorf, Verlag Data Becker.

Kröner, Peter (2010) HTML5 – Webseiten innovativ und zukunftssicher. München, Verlag Open Source Press.

Schober, Martin (2012) Mobil, mehrsprachig und multimedial – Grundlagen von HTML5, in technische kommunikation 34. Jahrgang Heft 6/12 S. 36-42, Stuttgart: Gesellschaft für technische Kommunikation e.V.

Vollendorf, Maximilian./ Bongers Frank. (2010): jQuery Das Praxishandbuch. Bonn, Galileo Press.

Wenz, Christian (2007) Javascript & AJAX. Das umfassende Handbuch. 7. Aufl. Bonn, Galileo Press.

Vollendorf, M./ Bongers F. (2010): jQuery Das Praxishandbuch. Bonn, Galileo Press.

Wenz, Christian (2007): Javascript & AJAX. Das umfassende Handbuch. 7. Aufl. Bonn, Galileo Press.

Wenz, Christian (2007) Javascript & AJAX. Das umfassende Handbuch. 7. Aufl. Bonn, Galileo Press. http://openbook.galileocomputing.de/javascript_ajax/

Wacker, S. (2007): selfhtml.org. SELFHTML e.V., Kiel

http://selfhtml.org/ und http://de.selfhtml.org/javascript/index.htm

http://www.html-world.de/program/ajax_2.php

http://jquery.com

http://jquerymobile.com/demos/1.0/

http://slides.html5rocks.com/#landing-slide