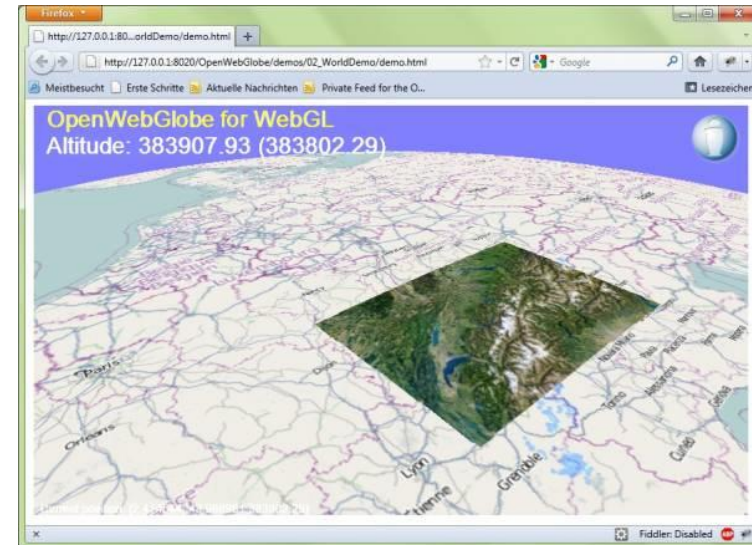
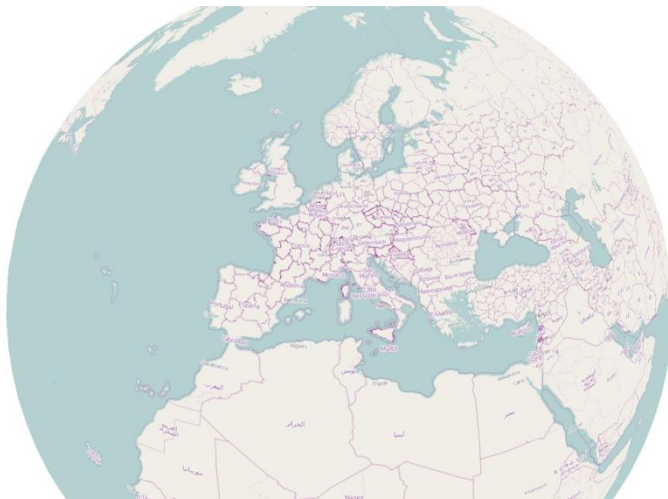


OpenWebGlobe SDK

An Open Source High-Performance Virtual Globe SDK for Open Maps



Martin Christen
University of Applied Sciences Northwestern Switzerland
martin.christen@fhnw.ch

Introduction: About OpenWebGlobe – Open Source Virtual Globe

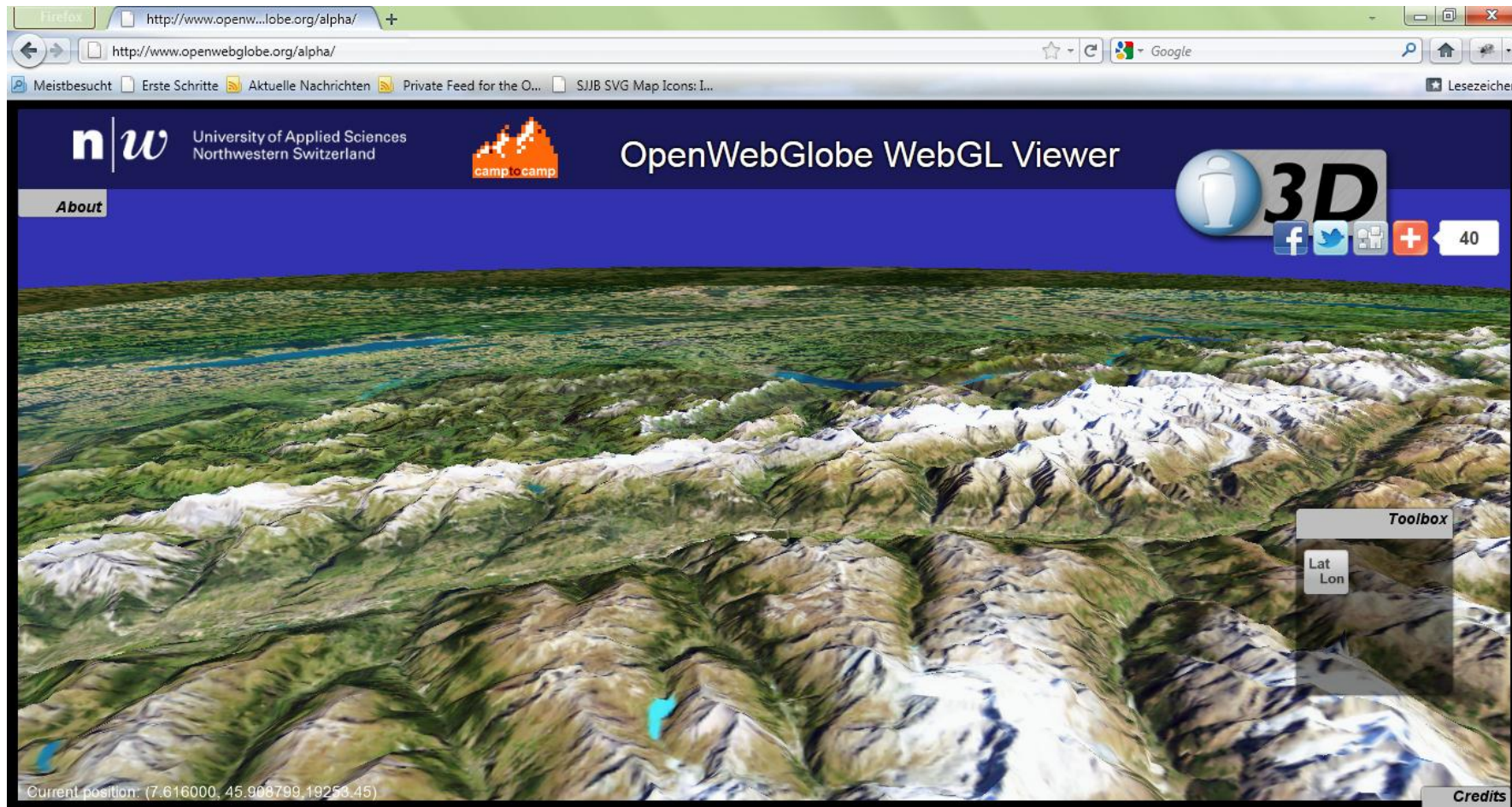


Videos on:
<http://www.openwebglobe.org>

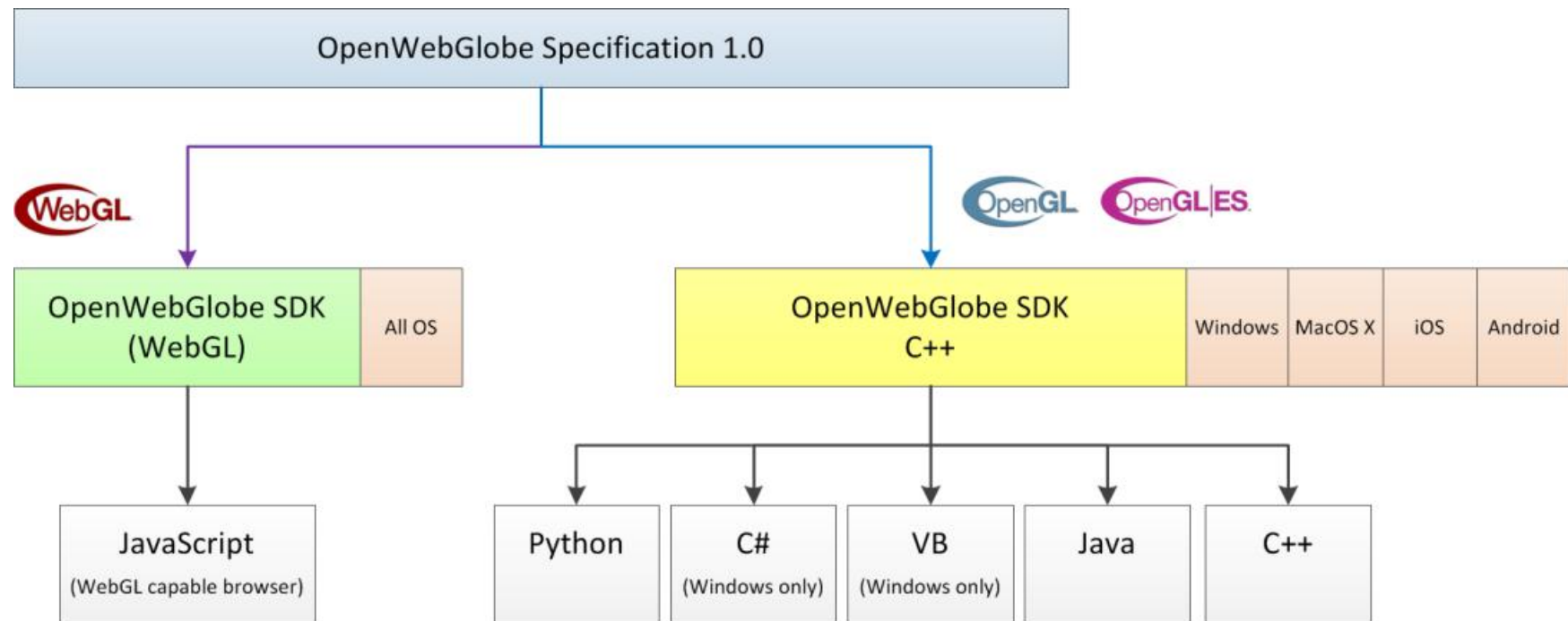


OpenWebGlobe in the Web Browser

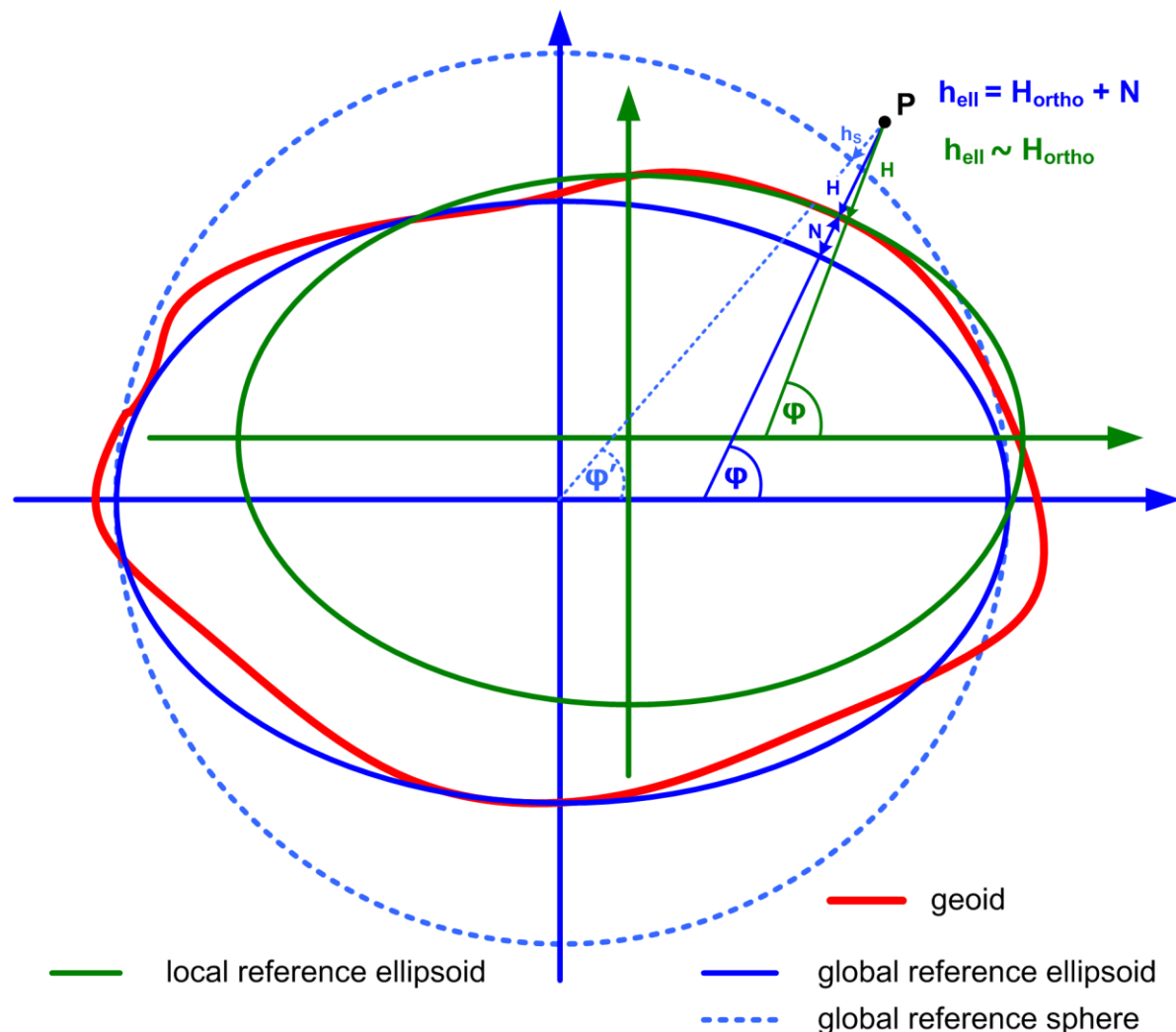
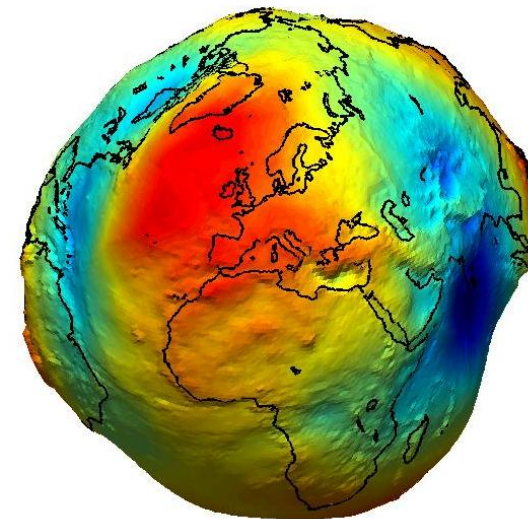
<http://www.openwebglobe.org/alpha>



OpenWebGlobe Technology



OpenWebGlobe – Geodetic Reference Systems



•Geodetic Reference Systems in Virtual Globes

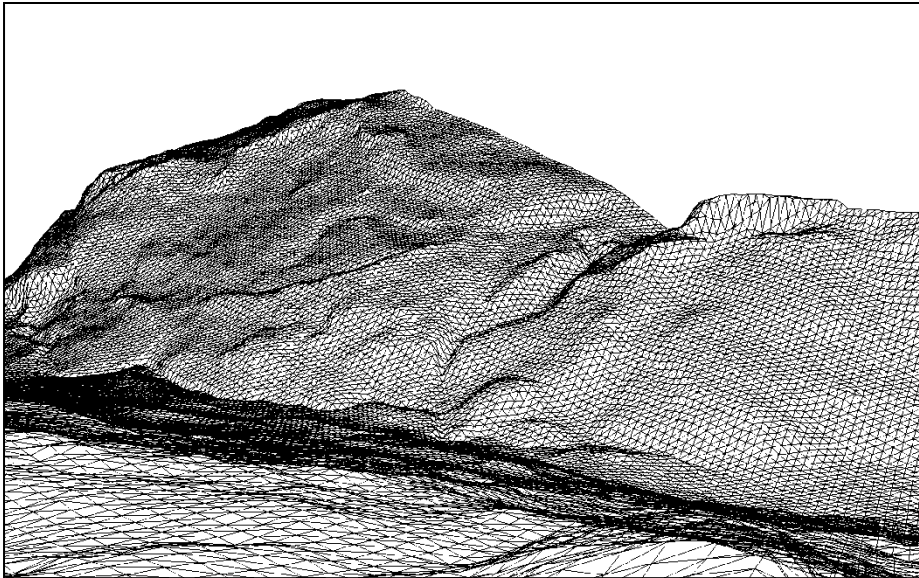
•Model in i3D

- WGS-84 ellipsoidal model &
- global geoid model

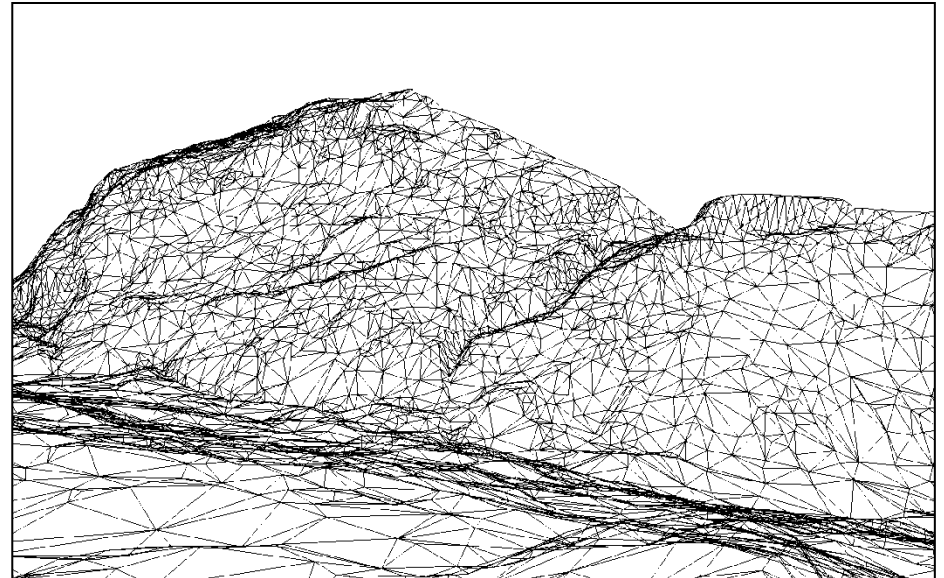
for high accuracies position
and height

Elevation Data Processing

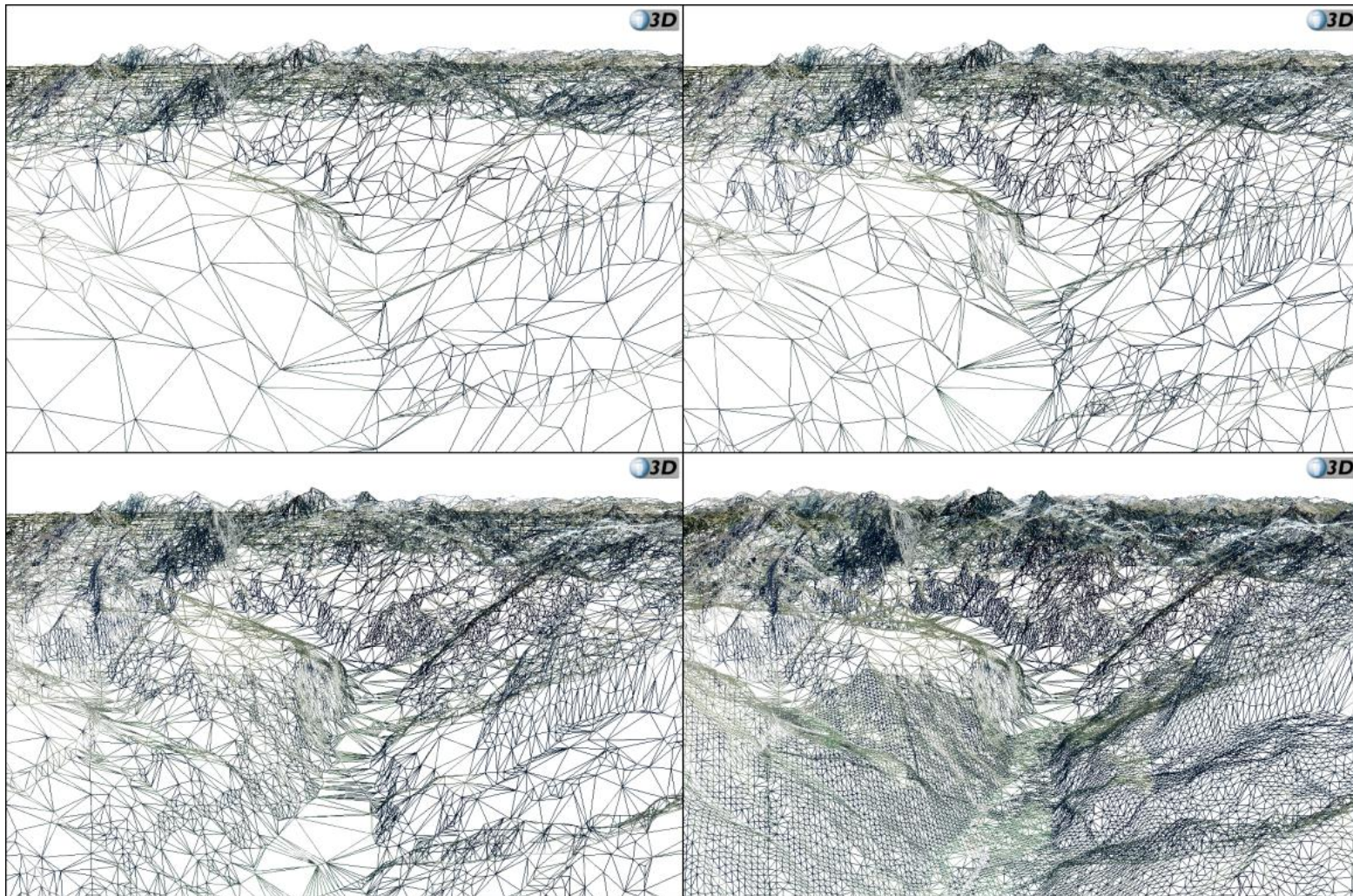
GRID



TIN



Elevation: Interactive Quality - from mobile devices to high end computers

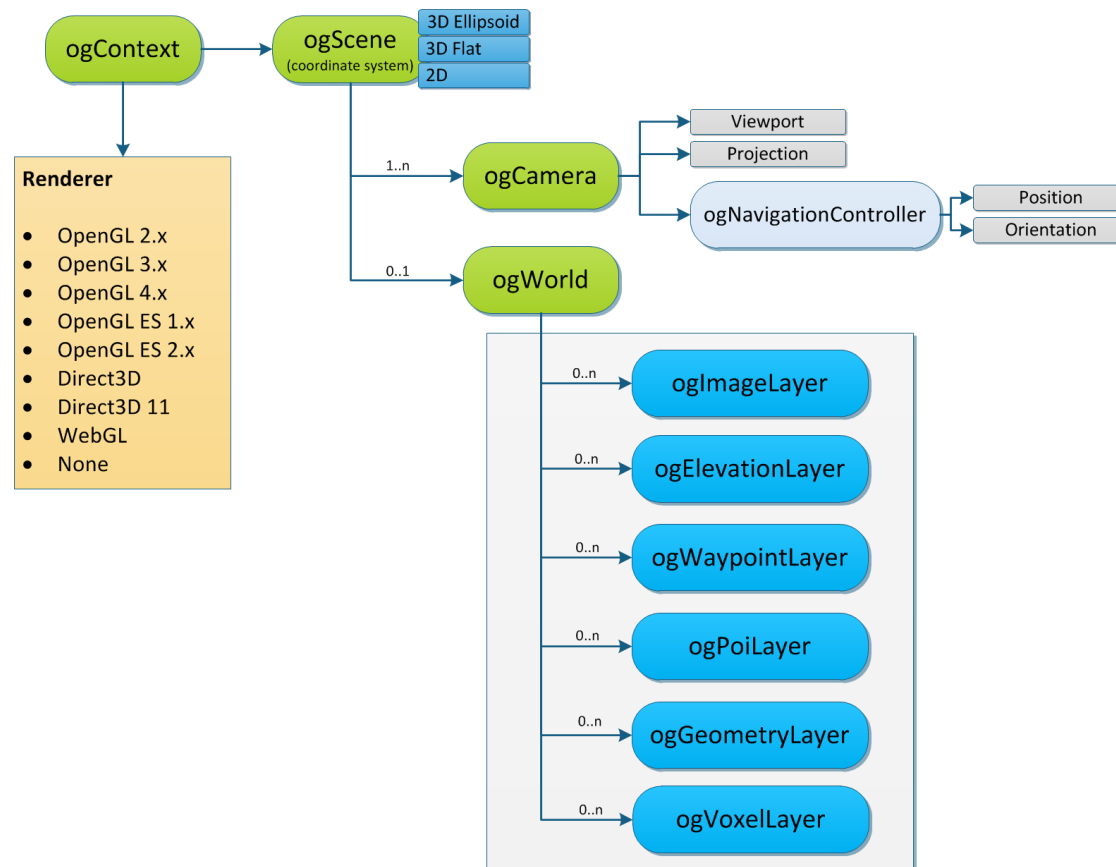


A Simple “Hello World” using the OpenWebGlobe SDK

```
context = ogCreateRenderWindow("Hello Globe", 640, 480);  
  
globe = ogCreateGlobe(context);  
  
ogAddImageLayer(globe, 'url: ["http://myglobe.com/img/"], service: "osm"');  
  
ogAddElevationLayer(globe, 'url: ["http://myglobe.com/elv/"], layer: "elv"');
```



OpenWebGlobe Objects



A simple «hello world» with OSM Data (JavaScript)

```
<script type="text/javascript" src="openwebglobe.js"></script>
<script type="text/javascript">

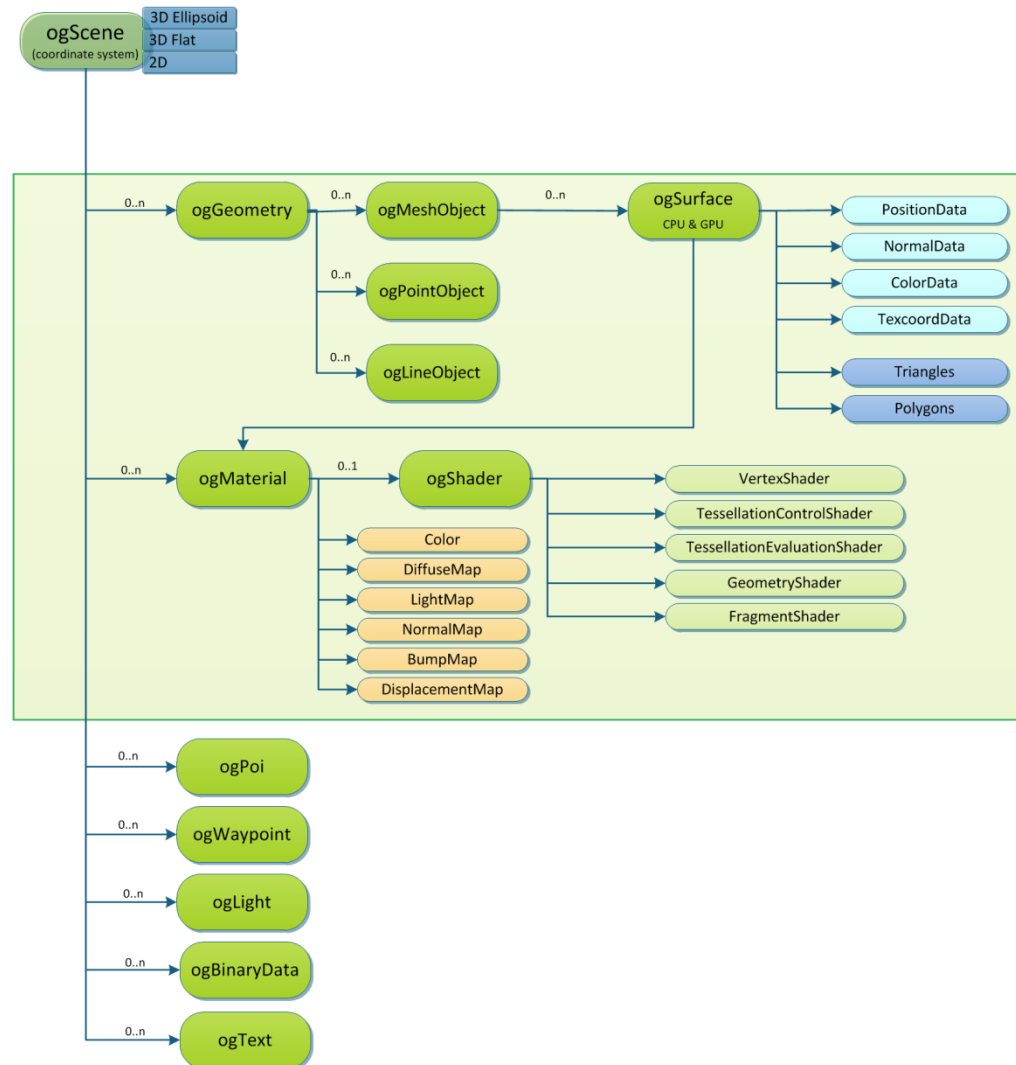
function main()
{
    var ctx = ogCreateContextFromCanvas("canvas", true);
    var globe = ogCreateGlobe(ctx);
    var imgOpenStreetMap =
    {
        url      : ["http://a.tile.openstreetmap.org",
                    "http://b.tile.openstreetmap.org",
                    "http://c.tile.openstreetmap.org"],
        service : "osm"
    };
    ogAddImageLayer(globe, imgOpenStreetMap);
}
</script>
<body onload="main()">
<div style="text-align: center">
<canvas id="canvas"></canvas></div></body>
```

A simple «hello world» with OSM Data: Adding POI



```
poilayer =
ogCreatePOILayer(globe,"airports",customTextstyle,customIconstyle);
var Poi =
{
    "icon"      : "http://.../airport.png",
    "text"      : "ZRH - Zürich",
    "position"  : [8.55, 47.4525, 1000]
};
ogCreatePOI(poilayer,Poi)
```

More OpenWebGlobe SDK Objects

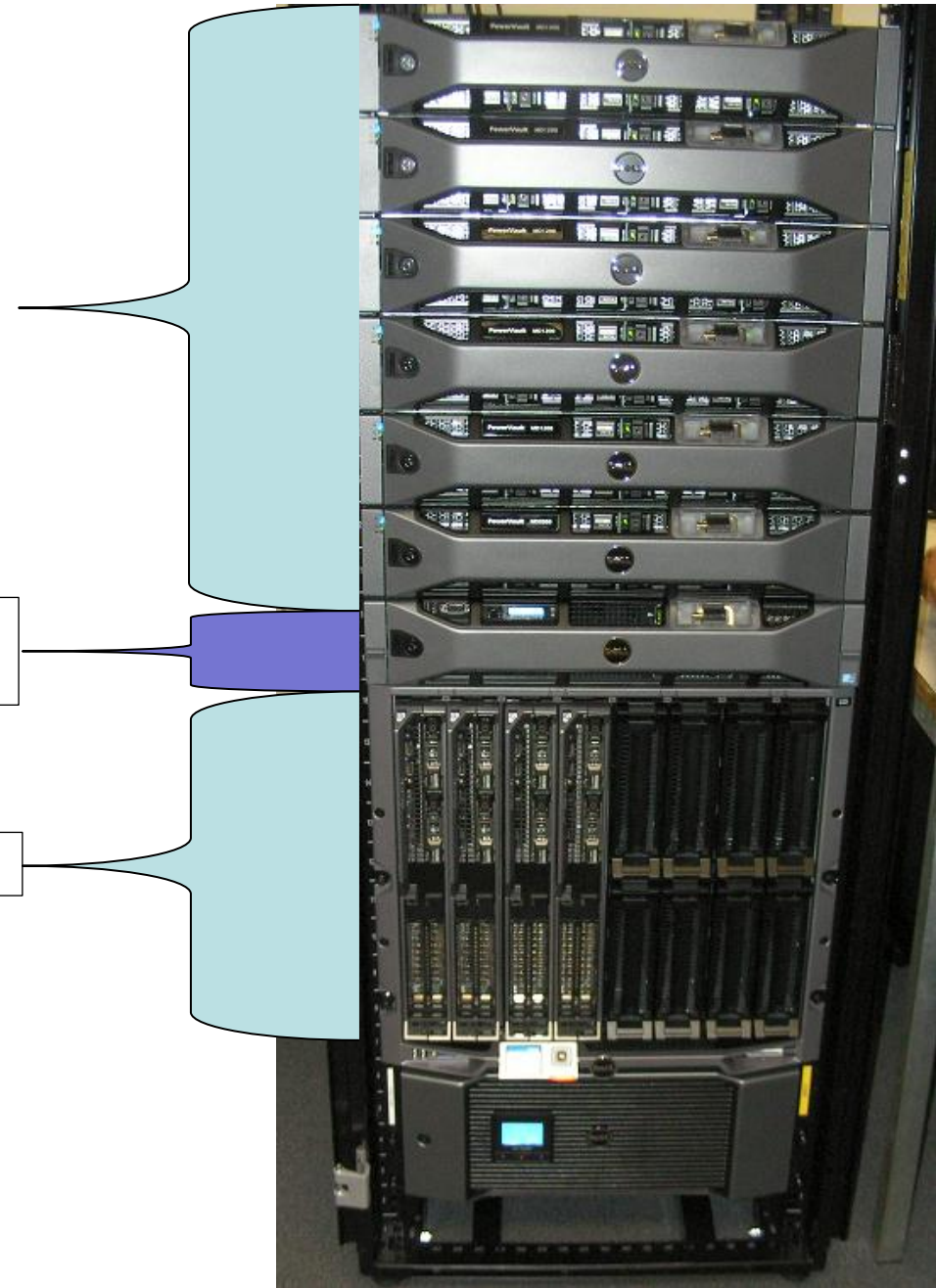


Data Processing

Storage: -120 Terabyte
 -1.4 GB/s Sequential Read
 -850 MB/s Sequential Write

Headnode: -
(Management of Compute Nodes)

Compute Nodes: - Data Processing



Data Processing Workflow

Create a Layer

Add Data to layer

Triangulate Data (only if elevation data)

Resample / Build all LOD

Update

Deploy

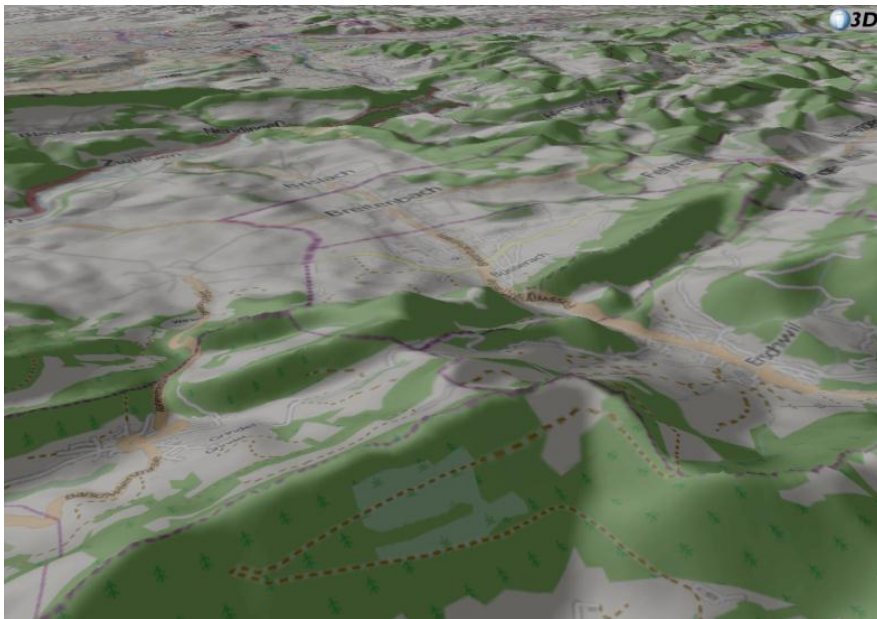
All Data Processing code is OpenSource
(MIT-License)

How we use OpenStreetMap Data



- **We render custom tiles: the «background» is transparent**
- **With transparent tiles we can combine different datasets**
- **We can combine it with orthophotos or by using hillshading tiles**

Combining OSM with hillshading and orthophotos



OpenWebGlobe on GitHub

- **The code is available on Github. OpenWebGlobe uses the «Closure Compiler» for JavaScript.**
- **A precompiled «SDK» is available together with some tutorials:**
 - **Text** (just draw some text)
 - **Texture** (just blit a texture)
 - **Globe** (display a simple globe)
 - **OpenStreetMap** (display globe with OSM data)
 - **Picking** (Click on Terrain to retrieve position including elevation)
 - **POI** (some examples with POI and custom styles)
 - **3D Geometry** (some examples of importing 3D-objects)
 - **Camera** (update position, orientation, have several cameras in a scene)
 - **Canvas as a billboard** (use HTML5 canvas as texture including video)

Coming soon: Point Cloud streaming support



Thank you!



OpenWebGlobe



OpenWebGlobe



martin.christen@gmail.com