

# OL3-CESIUM



## 3D for OpenLayers



<https://github.com/openlayers/ol3-cesium>

# Why, what, how?

# Why, what, how?

Add 3D to OpenLayers maps

OpenLayers can handle 3D data, Cesium can render it

Easy to learn for OpenLayers users

Federate efforts in one place

# Why, what, how?

Add 3D to OpenLayers maps

OpenLayers can handle 3D data, Cesium can render it

Easy to learn for OpenLayers users

Federate efforts in one place

Initially funded by a Boundless client

Created in cooperation with KlokanTech and Camptocamp

Additional funding by Camptocamp clients

# Why, what, how?

- Add 3D to OpenLayers maps
- OpenLayers can handle 3D data, Cesium can render it
- Easy to learn for OpenLayers users
- Federate efforts in one place

- Initially funded by a Boundless client
- Created in cooperation with KlokanTech and Camptocamp
- Additional funding by Camptocamp clients

- Initial release 1 year ago
- Camptocamp is actively working on new features
- Monthly releases

# Adding 3D to your map

# Adding 3D to your map

```
var ol3d = new olcs.OLCesium({map: map, target: id});  
ol3d.setEnabled(true);
```

# Adding 3D to your map

```
var ol3d = new olcs.OLCesium({map: map, target: id});  
ol3d.setEnabled(true);
```

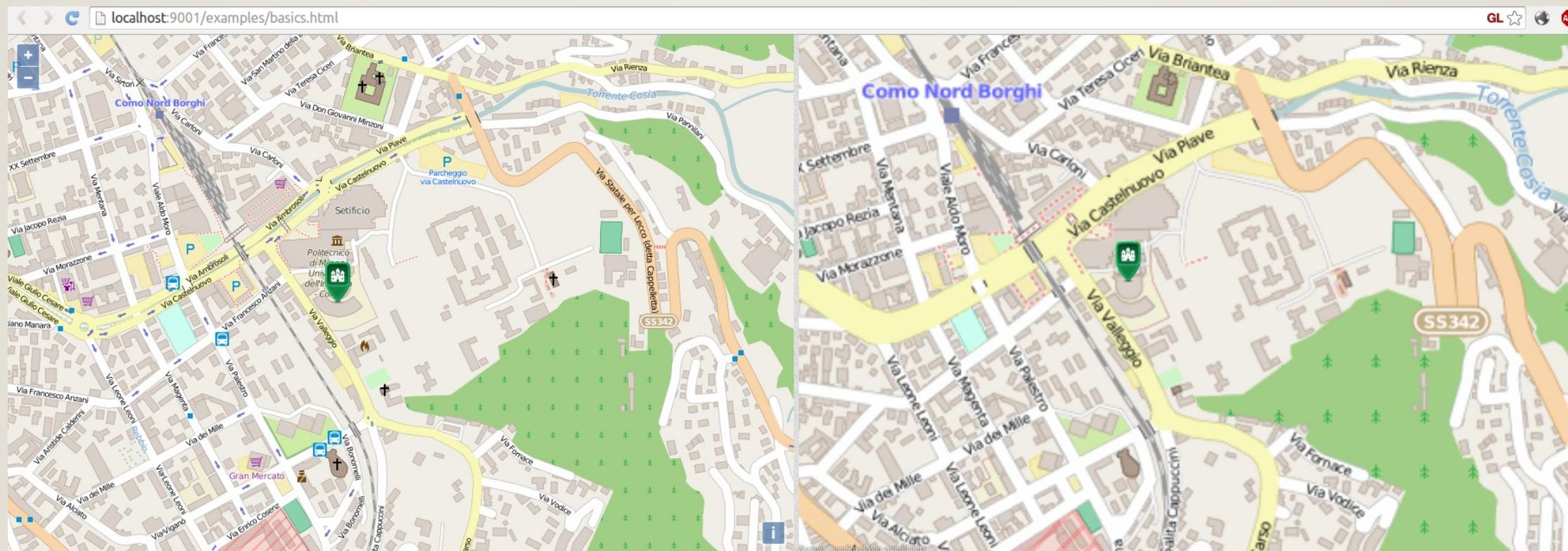
- A Cesium globe is created

# Adding 3D to your map

```
var ol3d = new olcs.OLCesium({map: map, target: id});  
ol3d.setEnabled(true);
```

- A Cesium globe is created
- layers and view are kept in sync

# Synchronized views



Try it

# 3D map with terrain

# 3D map with terrain

```
var ol3d = new olcs.OLCesium({map: map});
var scene = ol3d.getCesiumScene();
var terrainProvider = new Cesium.CesiumTerrainProvider({
    url: '//cesiumjs.org/stk-terrain/tilesets/world/tiles'
});
scene.terrainProvider = terrainProvider;
scene.globe.depthTestAgainstTerrain = true; // optional
ol3d.setEnabled(true);
```

# 3D map with terrain

```
var ol3d = new olcs.OLCesium({map: map});
var scene = ol3d.getCesiumScene();
var terrainProvider = new Cesium.CesiumTerrainProvider({
    url: '//cesiumjs.org/stk-terrain/tilesets/world/tiles'
});
scene.terrainProvider = terrainProvider;
scene.globe.depthTestAgainstTerrain = true; // optional
ol3d.setEnabled(true);
```

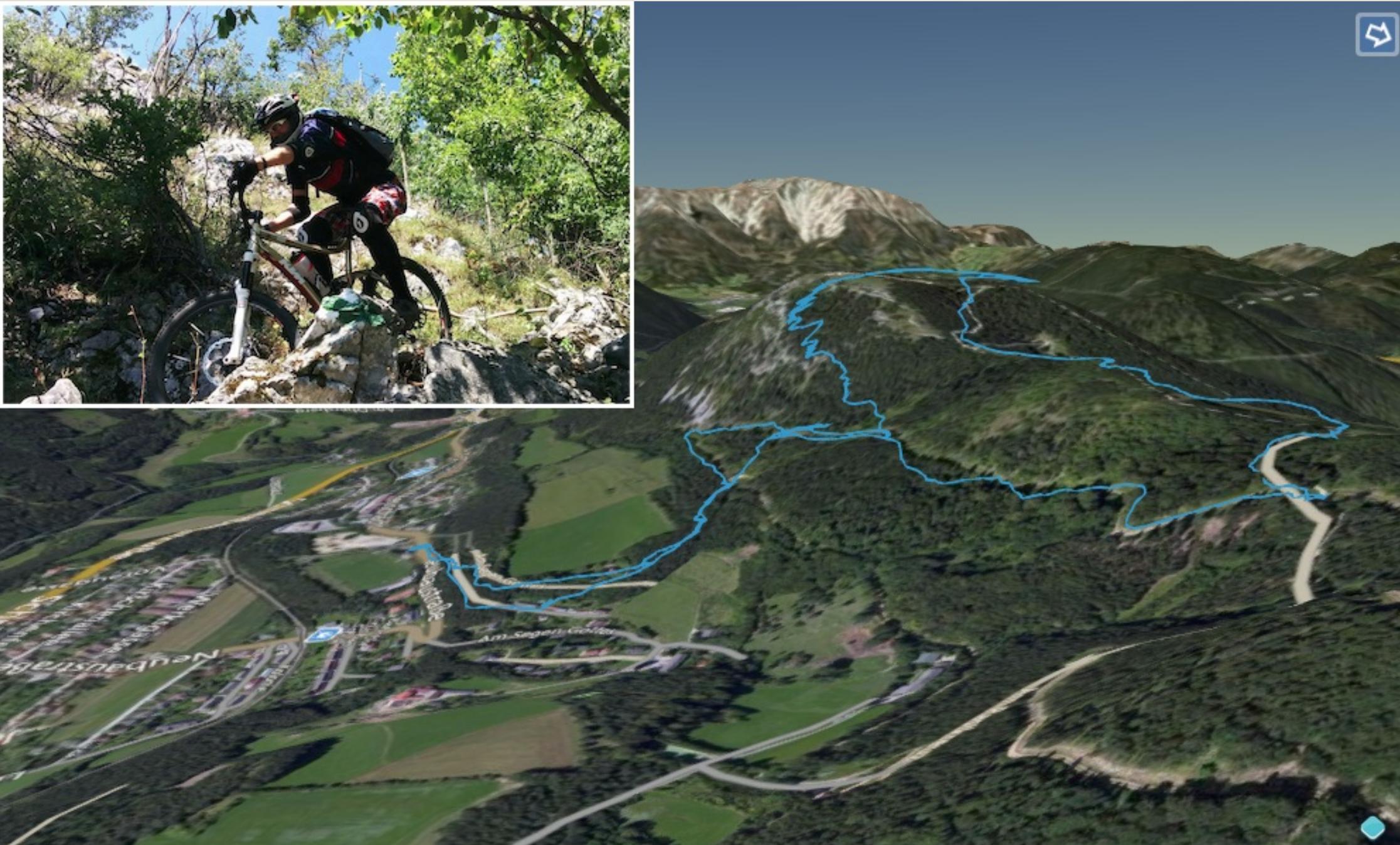
- Raster layers are clamped to ground

# 3D map with terrain

```
var ol3d = new olcs.OLCesium({map: map});
var scene = ol3d.getCesiumScene();
var terrainProvider = new Cesium.CesiumTerrainProvider({
    url: '//cesiumjs.org/stk-terrain/tilesets/world/tiles'
});
scene.terrainProvider = terrainProvider;
scene.globe.depthTestAgainstTerrain = true; // optional
ol3d.setEnabled(true);
```

- Raster layers are clamped to ground
- z-coordinates of vector data are used

# 3D GPS Track



OL3-Cesium – Guillaume Beraudo, Andreas Hocevar [Try it](#)

# Real life example

Suisse**Mobile** 3d

# Real life example

**SuisseMobile** 3d

- Swiss projection EPSG:21781

# Real life example

**SuisseMobile** 3d

- Swiss projection EPSG:21781
- Many layers including clusters

# Real life example

**SuisseMobile** 3d

- Swiss projection EPSG:21781
- Many layers including clusters
- Picking and tooltips

# Real life example

**SuisseMobile** 3d

- Swiss projection EPSG:21781
- Many layers including clusters
- Picking and tooltips
- Line and point features

# Real life example

**SuisseMobile** 3d

- Swiss projection EPSG:21781
- Many layers including clusters
- Picking and tooltips
- Line and point features
- Try it

# Different 2D projection

# Different 2D projection

- Cesium supports EPSG:4326 and EPSG:3857

# Different 2D projection

- Cesium supports EPSG:4326 and EPSG:3857
- Vector layers are reprojected automatically

# Different 2D projection

- Cesium supports EPSG:4326 and EPSG:3857
- Vector layers are reprojected automatically
- Raster layers must be handled by application

# Different 2D projection

- Cesium supports EPSG:4326 and EPSG:3857
- Vector layers are reprojected automatically
- Raster layers must be handled by application
  - No support for client side reprojection (yet?)

# Different 2D projection

- Cesium supports EPSG:4326 and EPSG:3857
- Vector layers are reprojected automatically
- Raster layers must be handled by application
  - No support for client side reprojection (yet?)
  - Require additionnal dataset in EPSG:4326 or EPSG:3857

# Vector clustering



# Vector clustering



- 30'000 points instead of reprojected rasters

# Vector clustering



- 30'000 points instead of reprojected rasters
- Pregenerated using a [custom OpenLayers3 tool](#)

# Vector clustering



- 30'000 points instead of reprojected rasters
- Pregenerated using a [custom OpenLayers3 tool](#)
- Picking: id, children and resolution per feature

# Vector clustering



- 30'000 points instead of reprojected rasters
- Pregenerated using a [custom OpenLayers3 tool](#)
- Picking: id, children and resolution per feature
- Data only sent once to the GPU, decimation in the shader

# 3D geometries



# 3D geometries



- Static positioning ( $x, y, z$ ) for all geometries

# 3D geometries



- Static positioning ( $x, y, z$ ) for all geometries
- Dynamic positioning (altitudeMode = "clampToGround")

# 3D geometries



- Static positioning ( $x, y, z$ ) for all geometries
- Dynamic positioning (altitudeMode = "clampToGround")
  - Points done

# 3D geometries



- Static positioning ( $x, y, z$ ) for all geometries
- Dynamic positioning (altitudeMode = "clampToGround")
  - Points done
  - Polygons coming soon

# 3D geometries

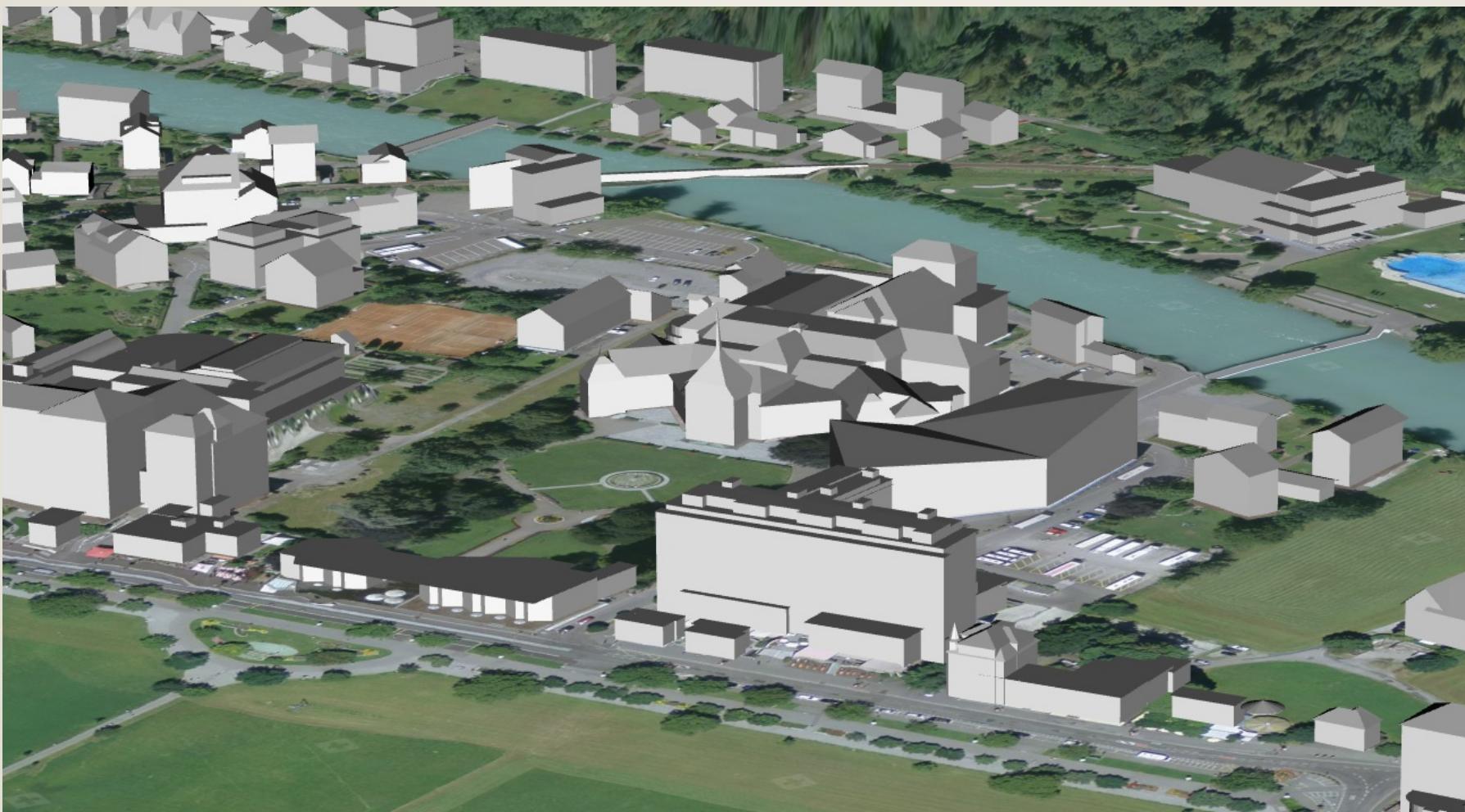


- Static positioning ( $x, y, z$ ) for all geometries
- Dynamic positioning (altitudeMode = "clampToGround")
  - Points done
  - Polygons coming soon
  - Lines are Work In Progress

# Buildings and vector tiles

# Buildings and vector tiles

- 3D buildings tile generation POC from CityGML



# Buildings and vector tiles

# Buildings and vector tiles

- 2.5D extruded polygons of a single "tile" POC



# Buildings and vector tiles

- 2.5D extruded polygons of a single "tile" POC



- Try it

# Buildings and vector tiles

# Buildings and vector tiles

- 3D-tiles specification is WIP in Cesium

# Buildings and vector tiles

- 3D-tiles specification is WIP in Cesium
  - Tiling and Levels Of Details

# Buildings and vector tiles

- 3D-tiles specification is WIP in Cesium
  - Tiling and Levels Of Details
  - Loading and unloading strategies

# Buildings and vector tiles

- 3D-tiles specification is WIP in Cesium
  - Tiling and Levels Of Details
  - Loading and unloading strategies
  - Efficient (GLTF)

# Questions?

openlayers / ol3-cesium

Unwatch 32 Star 69 Fork 26

OpenLayers - Cesium integration <http://openlayers.org/ol3-cesium/> — Edit

577 commits 5 branches 9 releases 9 contributors

Branch: master ol3-cesium / +

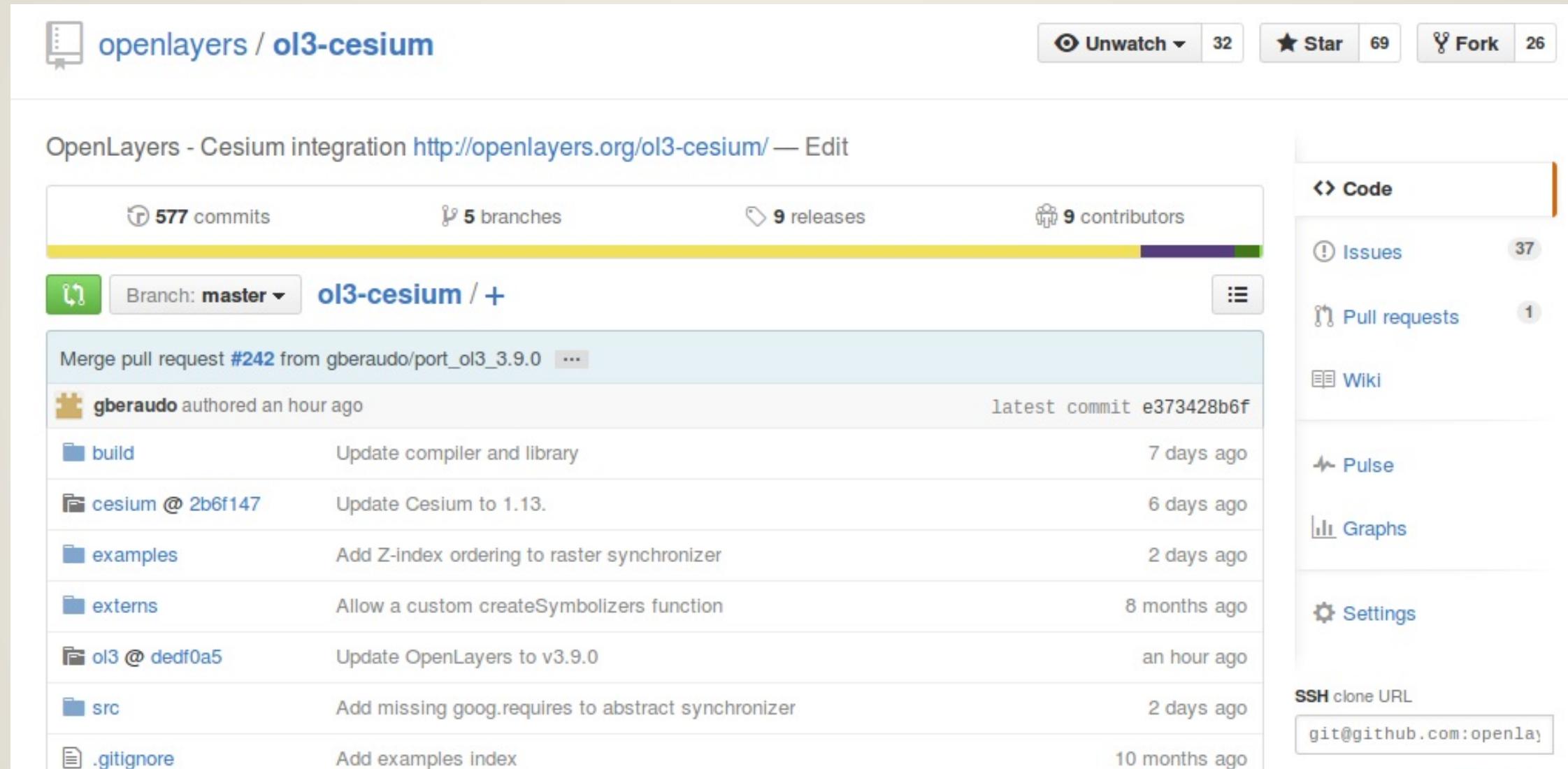
Merge pull request #242 from gberaudo/port\_ol3\_3.9.0 ...

gberaudo authored an hour ago latest commit e373428b6f

File	Description	Time
build	Update compiler and library	7 days ago
cesium @ 2b6f147	Update Cesium to 1.13.	6 days ago
examples	Add Z-index ordering to raster synchronizer	2 days ago
externs	Allow a custom createSymbolizers function	8 months ago
ol3 @ dedf0a5	Update OpenLayers to v3.9.0	an hour ago
src	Add missing goog.requires to abstract synchronizer	2 days ago
.gitignore	Add examples index	10 months ago

Code Issues Pull requests Wiki Pulse Graphs Settings

SSH clone URL git@github.com:openla...



감사합니다

OL3-Cesium – Guillaume Beraudo, Andreas Hocevar