

Hacking with Earth Engine and Google Earth

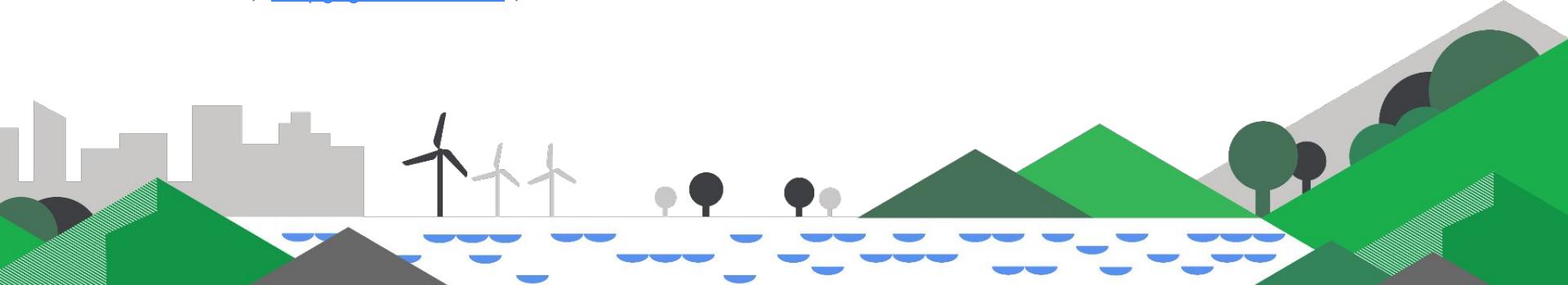
fun with JavaScript & KML



Christiaan Adams

Developer Relations, Google Earth Outreach

October 2022 | bit.ly/g4g22-hackearth | #GeoForGood22



Your Hosts:



Christiaan Adams

Developer Relations Engineer,
Google Earth Outreach

California, USA

Expertise: Google Earth, KML, GIS,
Imagery,



Gennadii "Gena" Donchyts

Customer Engineer,
Google Earth Engine,
Google Cloud / Geo Enterprise

Netherlands

Expertise: Google Earth Engine,
Remote Sensing, GIS, AI/ML



David Tryse

KML Hacker Extraordinaire,
Google Earth & Maps consultant

Ireland

Expertise: Google Earth, KML,
Earth Engine, BigQuery

#GeoForGood22

Agenda

01 Earth Engine hacks

Fun ways to visualize things and create new datasets

03 Google Earth hacks

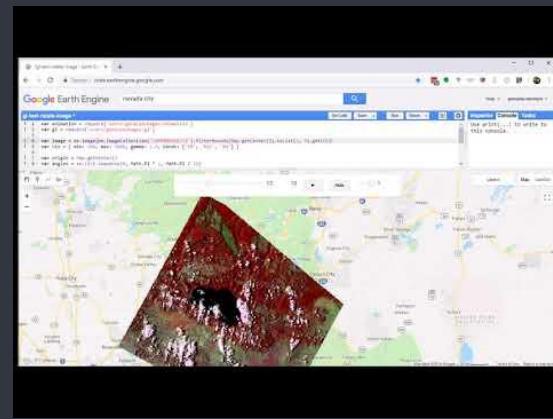
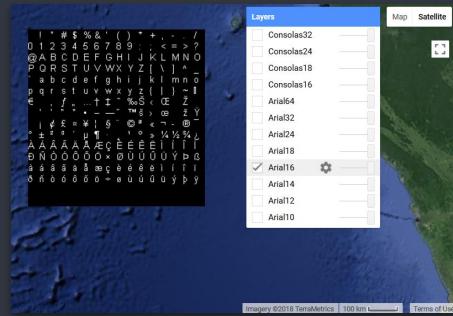
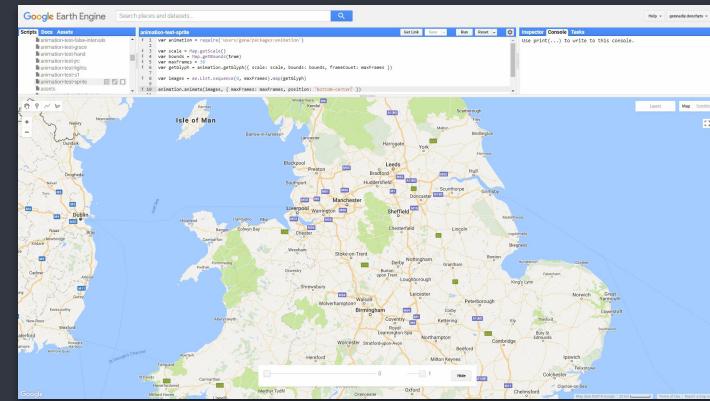
Dynamic storytelling and more...
Did you know you can do that?

02 Bringing Maps from EE into Earth

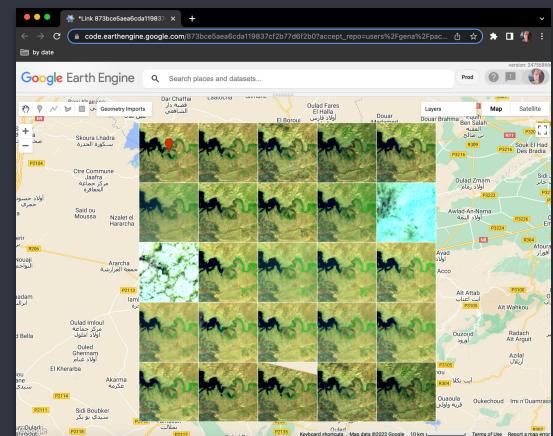
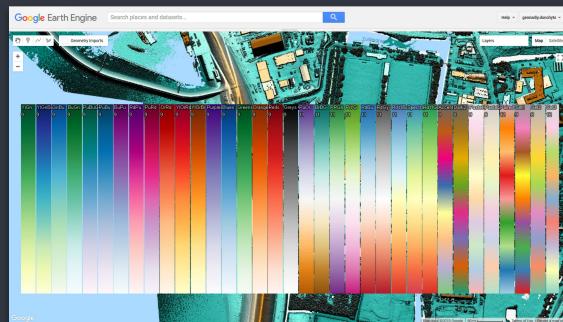
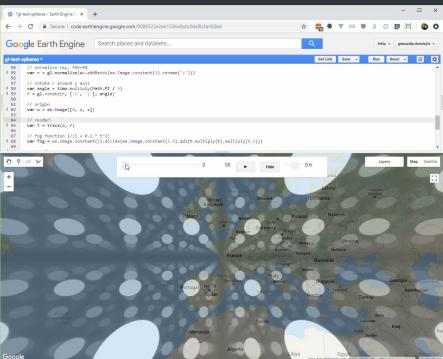
Buckets and tiles

04 Question & Answer

How do I... ?



Google Earth Engine Hacks



Outline

animation - images, indices, intervals, grace, hand, jrc, lights, s1/ice/sm, sprite, temperature

gallery - image collection → image

text - simple, annotate, fonts

palette - Color Brewer, cmocean

gl - pretty much useless, but cool! :), rotate image, mesh, sphere, ray marching, dem

assets - merge, group, clean, CDF burst

style - dark map, frame, scalebar, gradientbar, north arrow

charting - scatter, radiance/reflectance

hydro - catchments, rivers, dams, water occurrence (JRC), flow accumulation (HydroSHEDS) dem/hand

utils - iso, hex, norm, profile, ratio, fast min/max, rgb3d, radians, rescale, pad, PM, skeletonize, HoM, despeckle(t)

[Scripts](#) [Docs](#) [Assets](#)

- users/gena/packages
 - animation
 - animation-test-false
 - animation-test-false-indices
 - animation-test-false-intervals
 - animation-test-false-intervals-S3
 - animation-test-grace
 - animation-test-hand
 - animation-test-jrc
 - animation-test-lights
 - animation-test-s1
 - animation-test-s1-SM
 - animation-test-s1-by-day
 - animation-test-s1-ice
 - animation-test-sprite
 - animation-test-temperature
 - assets
 - assets-test-burn-bad-pixels
 - assets-test-cloudfree.js
 - assets-test-filterclouds
 - assets-test-get-by-name
 - chart
 - charting
 - charting-test-radiance-reflectance
 - charting-test-simple
 - cloud-utils
 - colorbrewer
 - colorbrewer-test
 - colorbrewer-test-custom
 - colorbrewer-test-custom-slic
 - gallery
 - gallery-test
 - geometry
 - geometry-test-transect
 - gl
 - gl-test-dem
 - gl-test-empty
 - gl-test-mesh
 - gl-test-rotate-image
 - gl-test-sphere

<https://github.com/gee-community/ee-packages-js>

Share Repo: users/gena/packages

gennadiy.donchys@gmail.com

Owner

Email or domain

Reader

Add

 Anyone can read

Users who have access to the project will be able to access it in the Code Editor via this link:

https://code.earthengine.google.com/?accept_repo=users/gena/packages

Or they may clone its Git repository by running the following command in a terminal:

```
git clone https://earthengine.googlesource.com/user/
```

[Done](#)https://code.earthengine.google.com/?accept_repo=users/gena/packages

animation

images, features, indices, composites, sprites, water, lights, temperature ...



```
var animation = require('users/gena/packages:animation')

var images = ee.ImageCollection('COPERNICUS/S2')
  .filterBounds(geometry)

var vis = { min: 300, max: 4000, bands: ['B12', 'B8', 'B3'] }

animation.animate(images, {
  vis: vis,
  position: 'bottom-right',
  maxFrames: 50,
  width: '500px'
})
```

animation

sprites

Google Earth Engine Help gennadiy.donchys ▾

Scripts Docs Assets

- animation-test-false-intervals
- animation-test-grace
- animation-test-hand
- animation-test-jrc
- animation-test-flights
- animation-test-s1
- animation-test-sprite
- assets

animation-test-sprite

```
i 1 var animation = require('users/gena/packages:animation')
i 2
i 3 var scale = Map.getScale()
i 4 var bounds = Map.getBounds(true)
i 5 var maxFrames = 30
i 6 var getGlyph = animation.getGlyph({ scale: scale, bounds: bounds, frameCount: maxFrames })
i 7
i 8 var images = ee.List.sequence(0, maxFrames).map(getGlyph)
i 9
i 10 animation.animate(images, { maxFrames: maxFrames, position: 'bottom-center' })
```

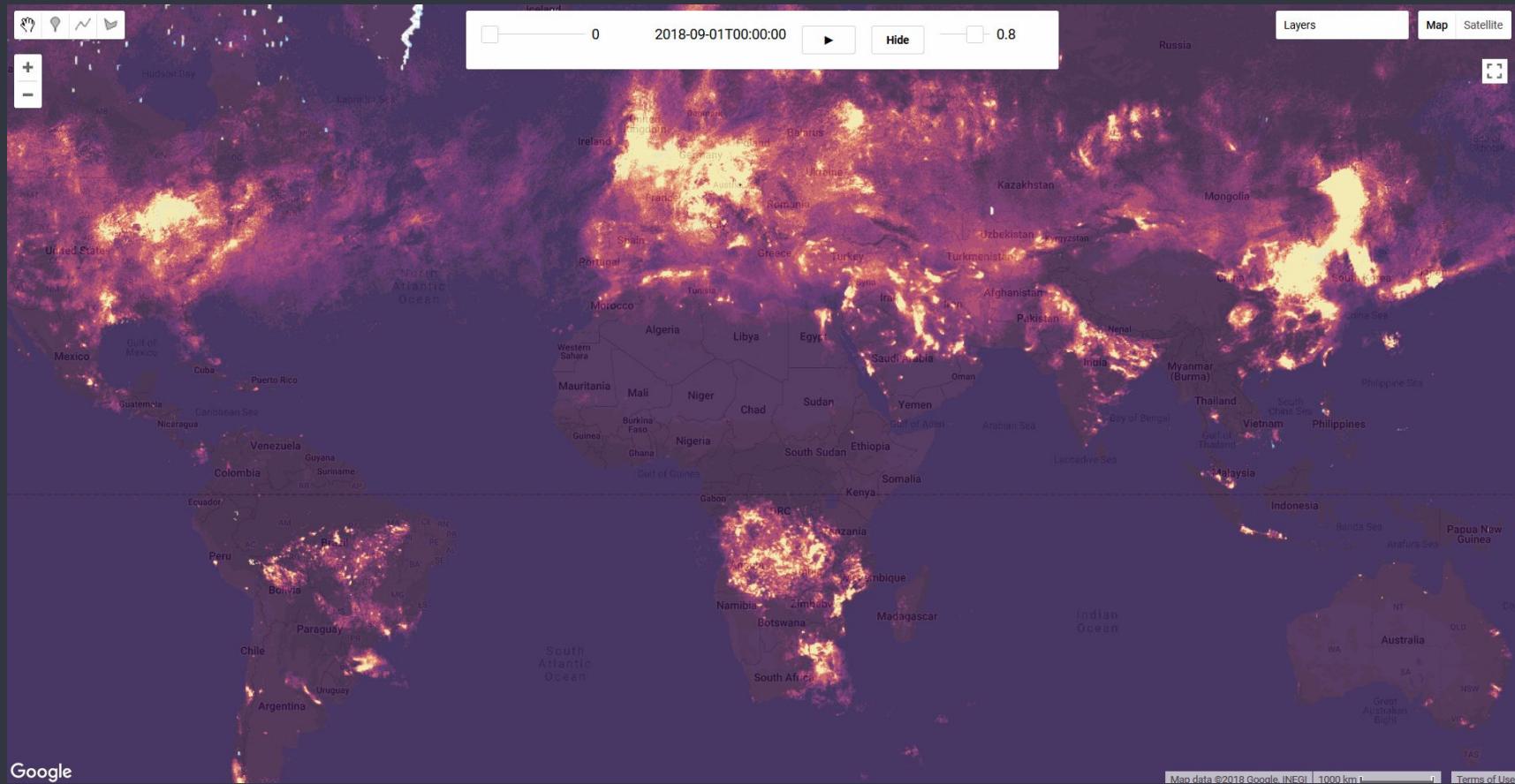
Get Link Save Run Reset ⚙️ Inspector Console Tasks

Use print(...) to write to this console.

Layers Map Satellite

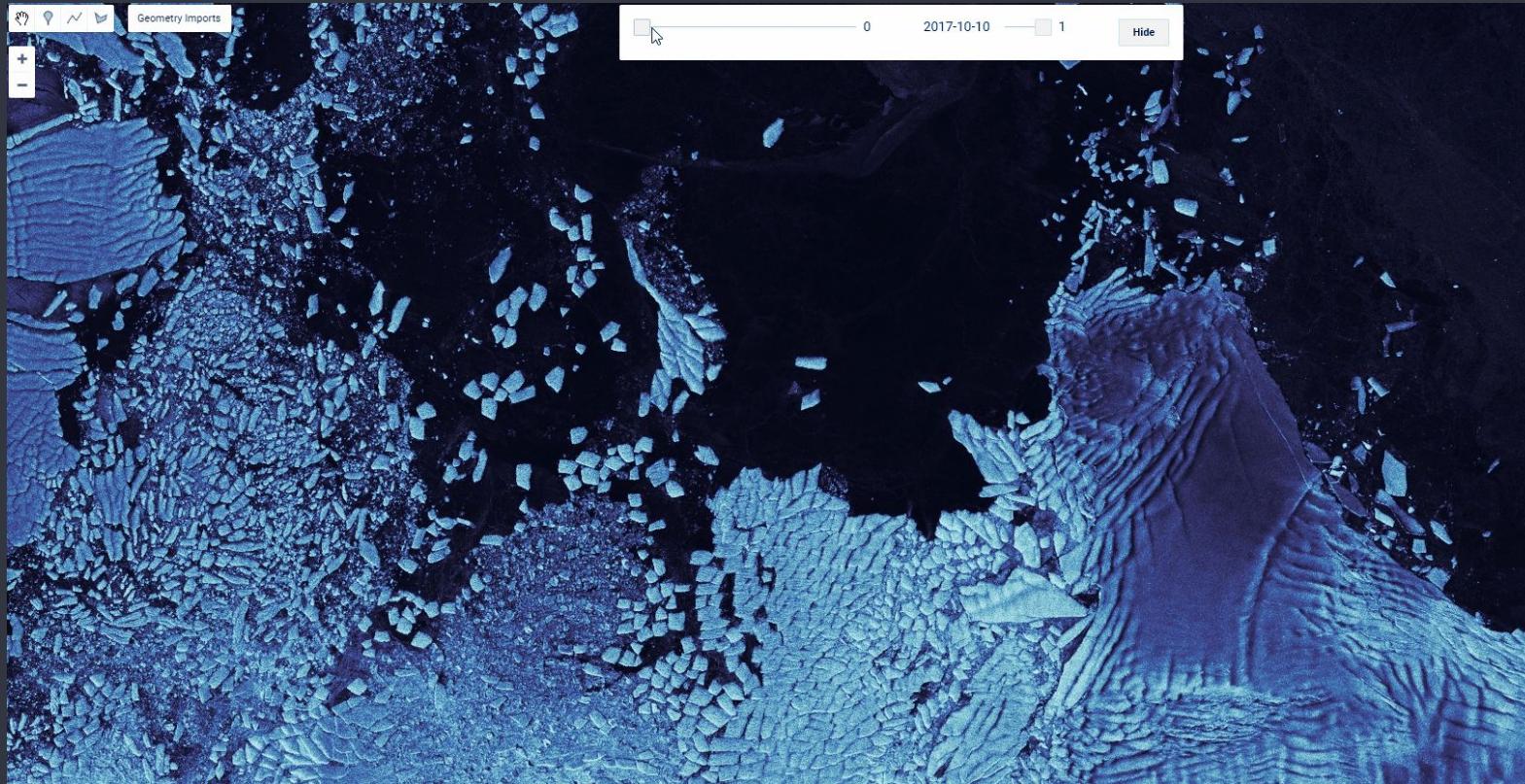
Code

animation



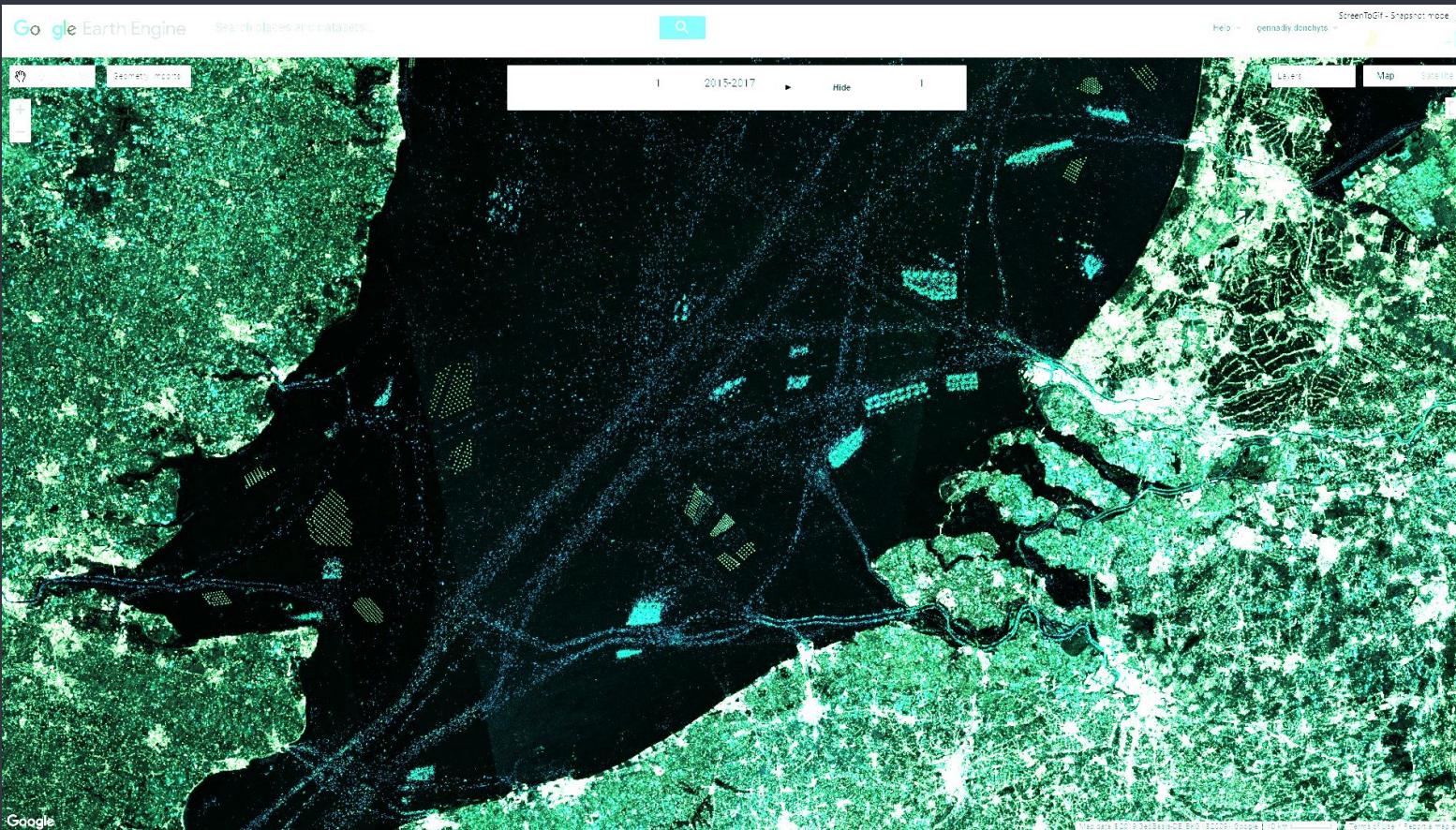
animation

ESA Sentinel-1, ice movement



[Code](#)

animation



gallery

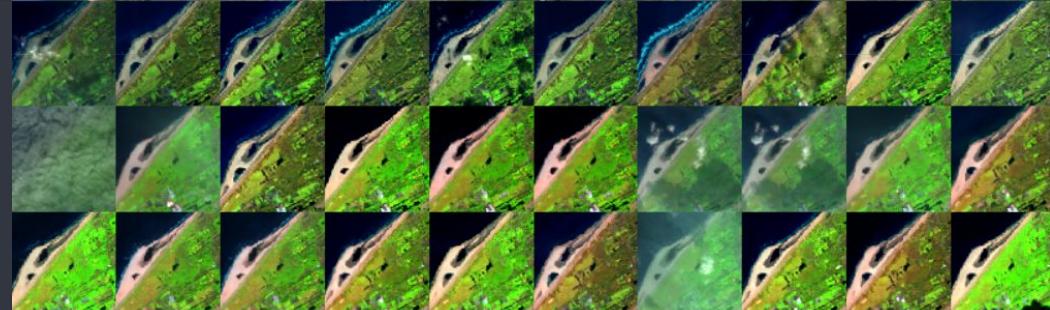
image collection → image



```
var gallery = require('users/gena/packages:gallery')

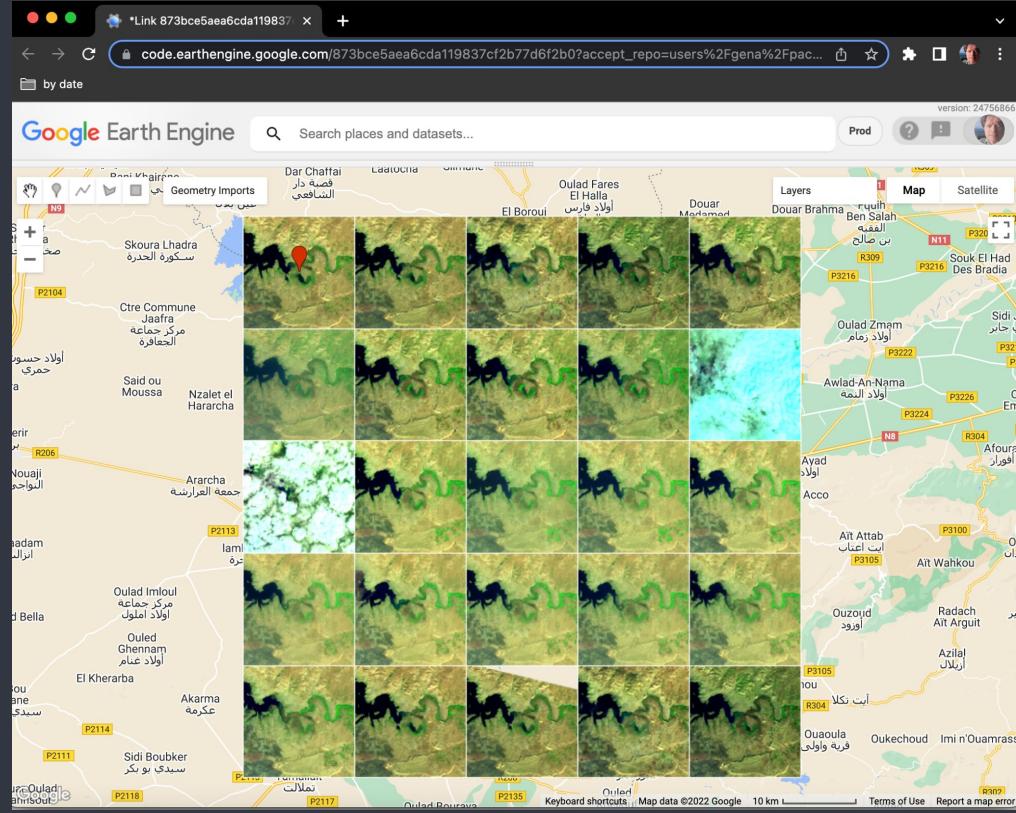
// ...

var imageGallery = gallery.draw(images, region, rows, columns)
```



gallery

image collection → image



Code

palette

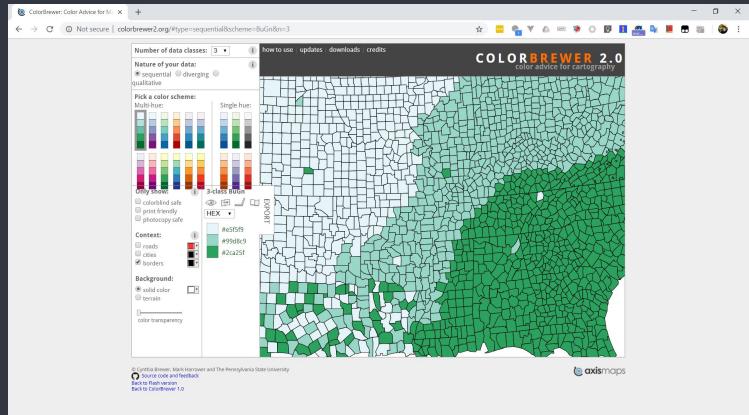
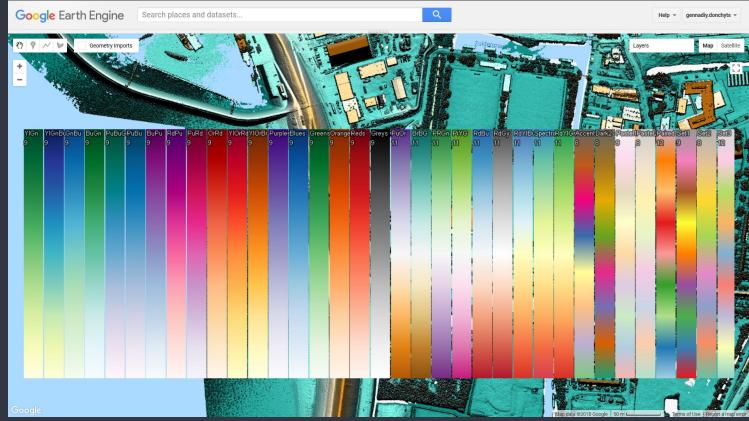
Color Brewer, cmocean



```
var palettes = require('users/gena/packages:palettes')

var palette1 = palettes.colorbrewer.Paired[7]

var palette2 = palettes.cmocean.Thermal[7]
```

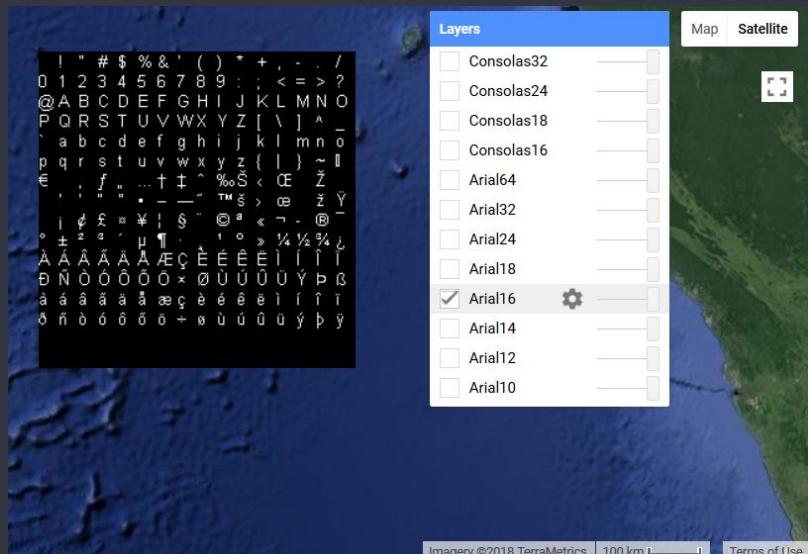


<https://github.com/gee-community/ee-palettes>

text

simple, annotate, fonts

```
● ● ●  
var text = require('users/gena/packages:text')  
  
// draw text at map center  
var str = 'Allo, how are you?'  
var point = Map.getCenter()  
var text = text.draw(str, point, scale, { fontSize: 32 })  
  
// add the text image as a raster layer  
Map.addLayer(text, {}, 'simple text')
```

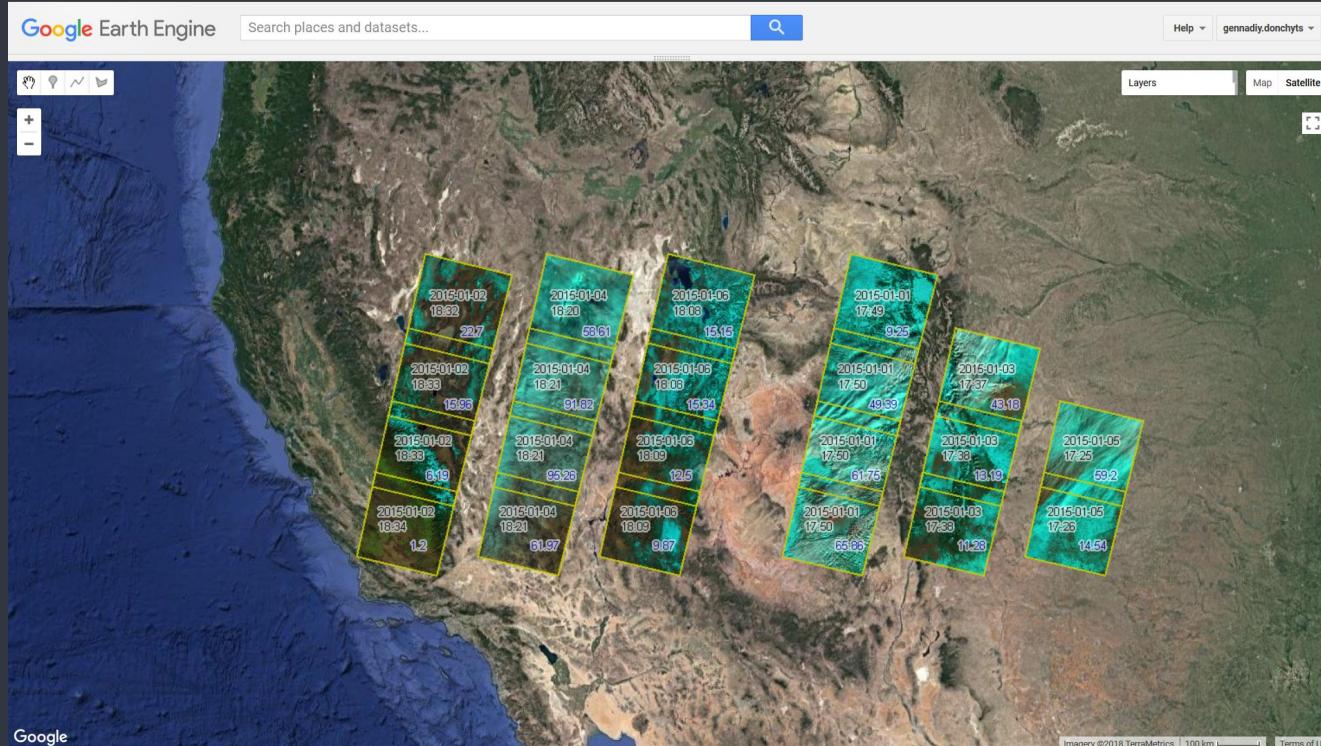


String → ASCII → row, column → width → translate

Code

text

simple, annotate, fonts

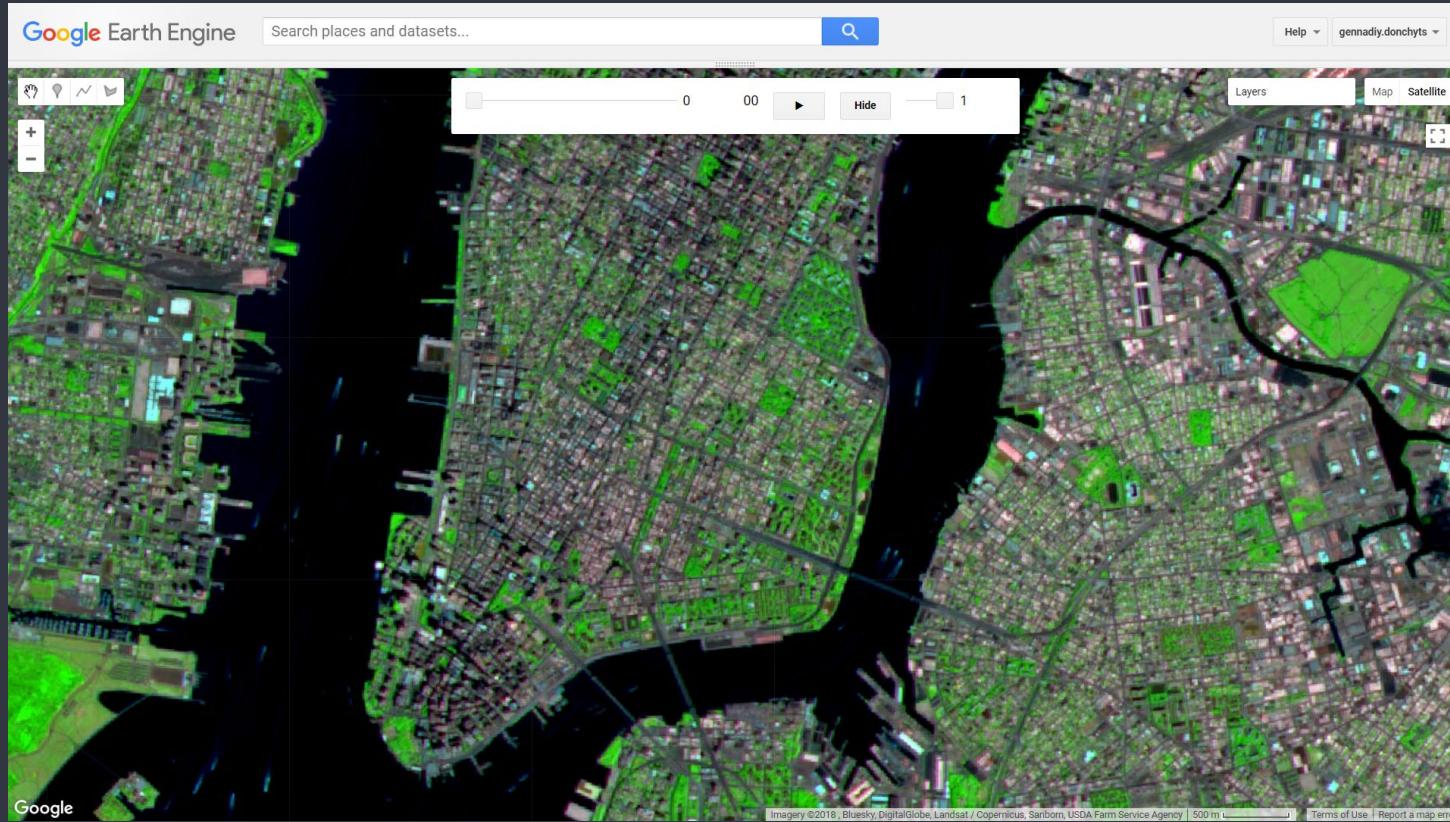


[Code](#)

gl

experimental, rotate image, ray marching, dem

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

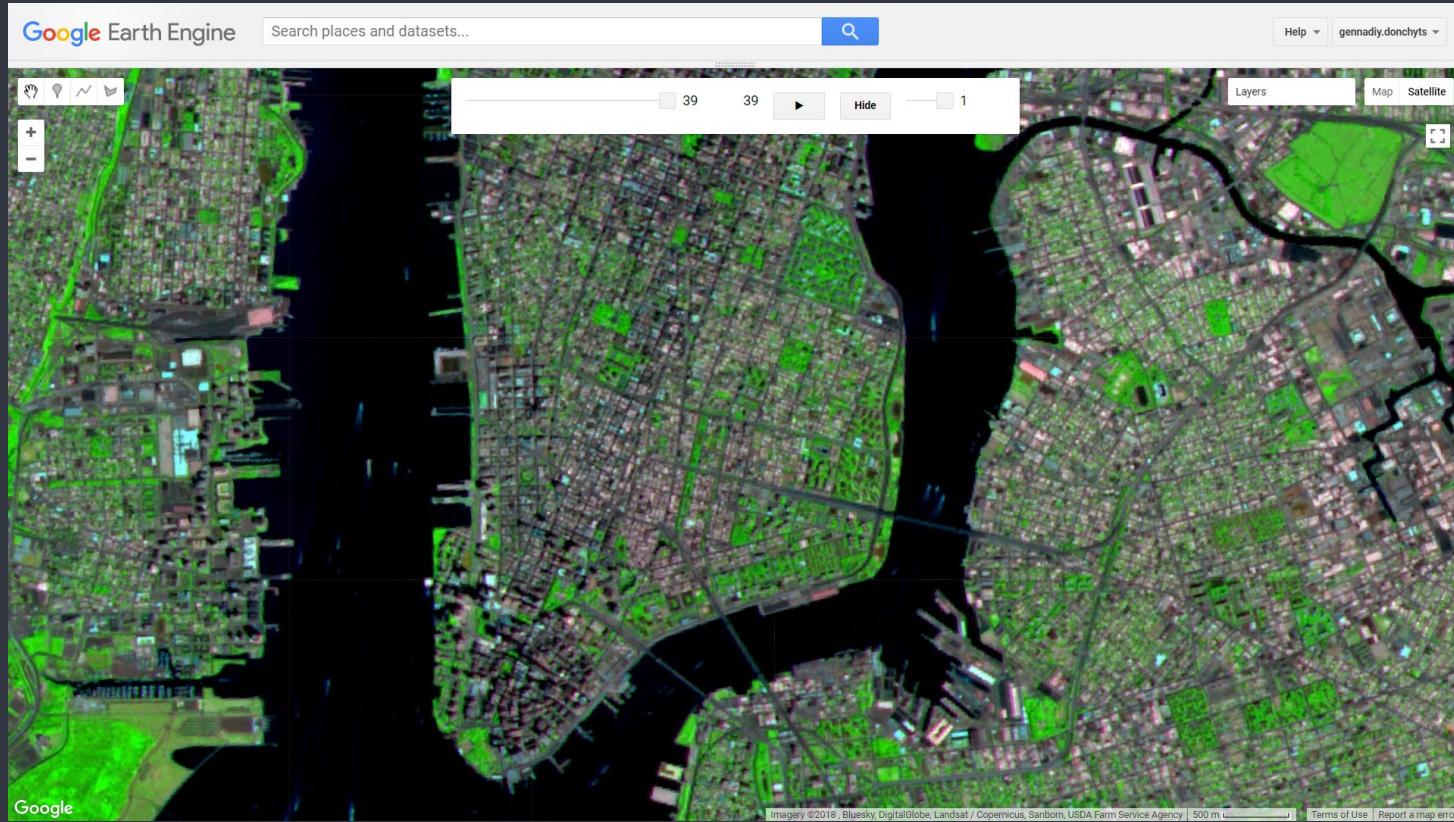


[Code](#)

gl

experimental, rotate image, ray marching, dem

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

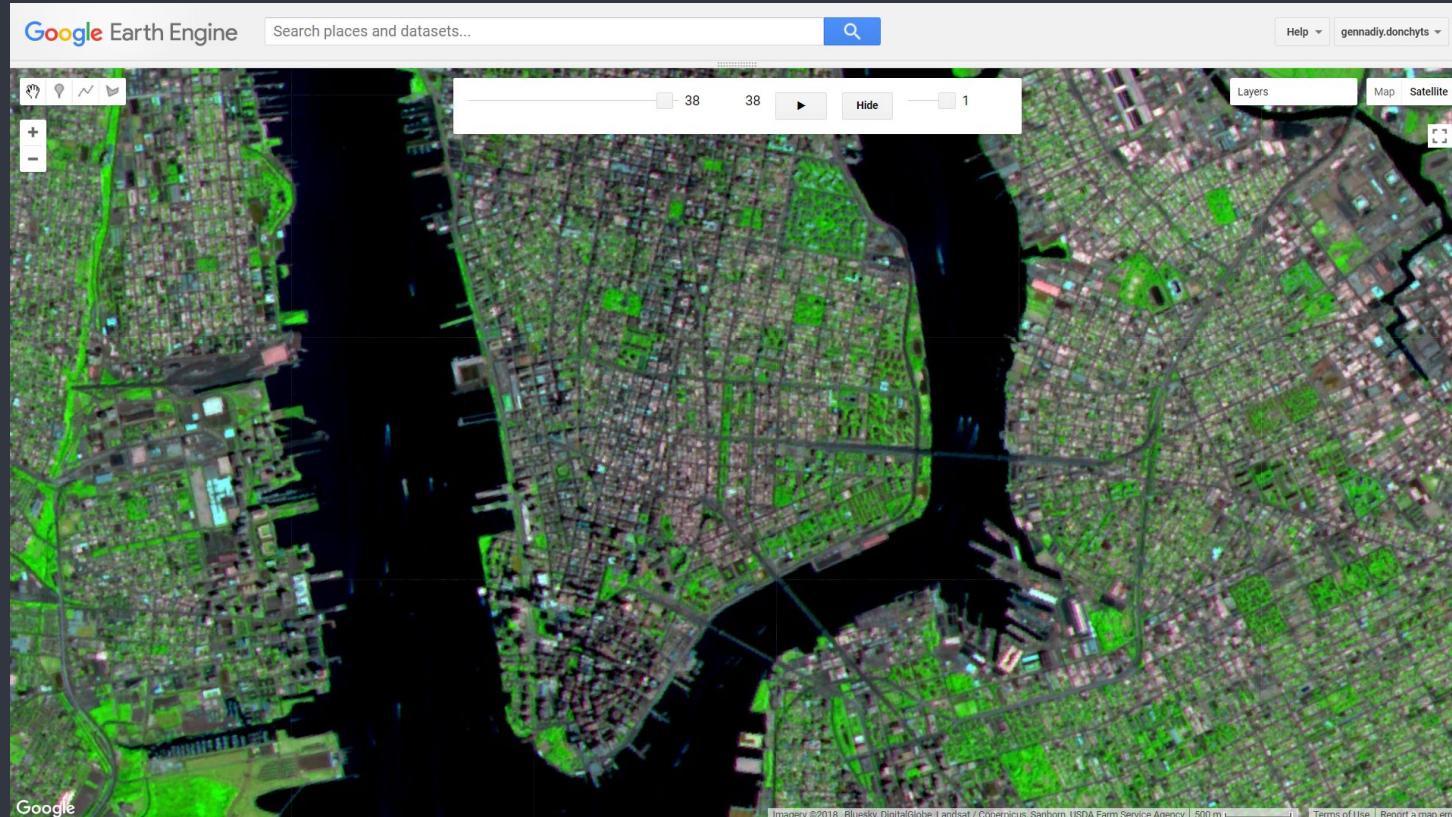


[Code](#)

gl

experimental, rotate image, ray marching, dem

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos\theta & -\sin\theta \\ \sin\theta & \cos\theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$

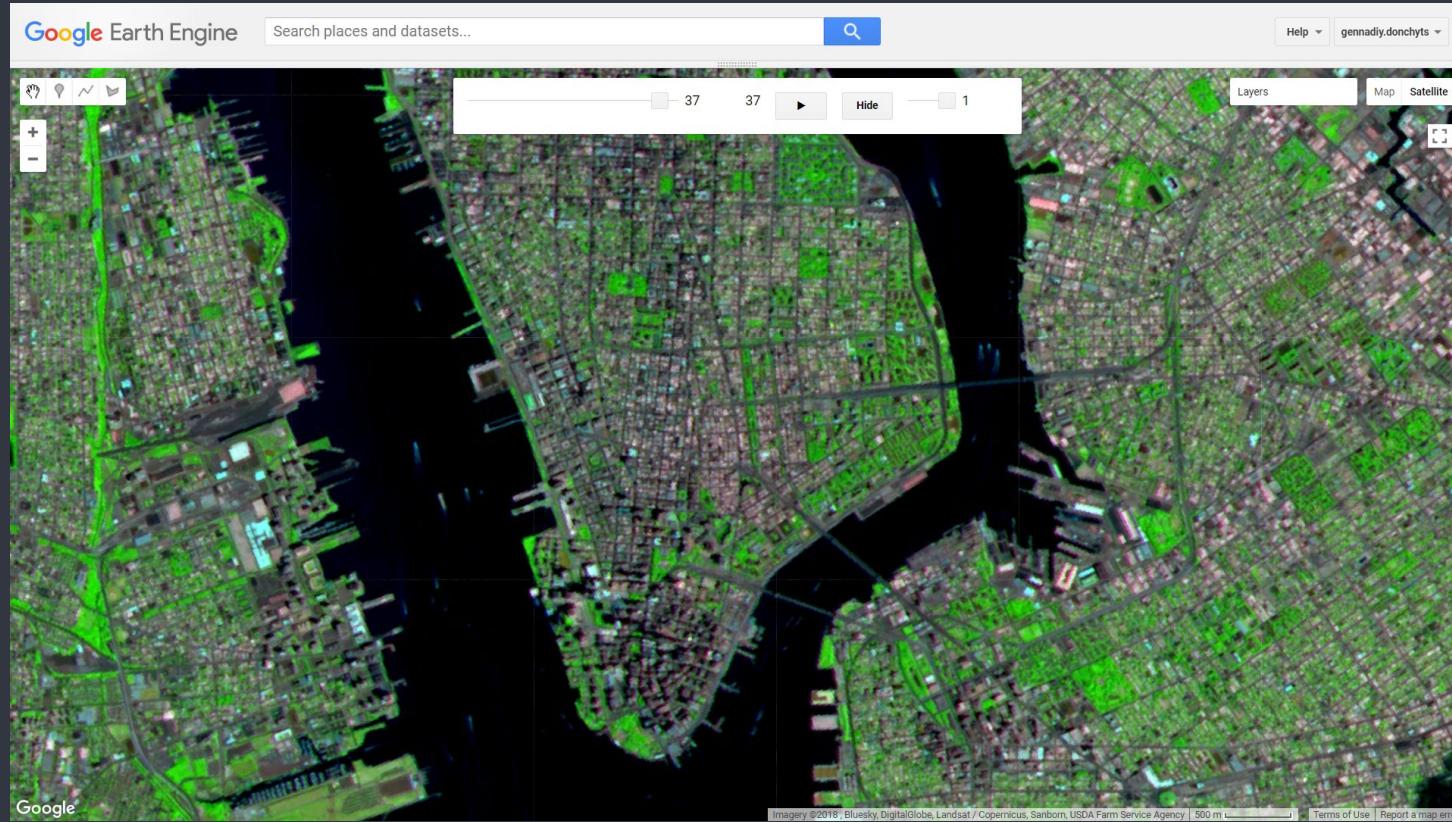


Code

gl

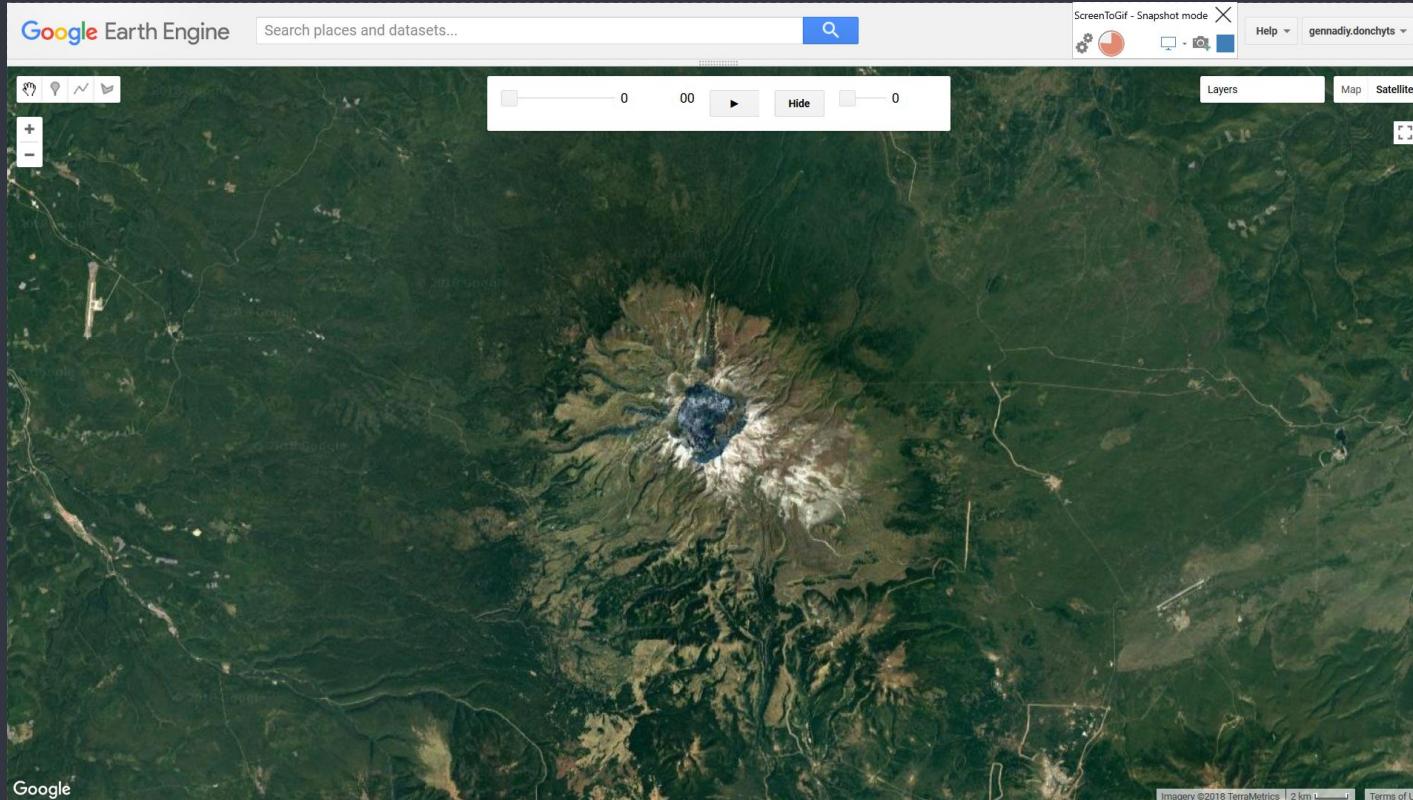
experimental, rotate image, ray marching, dem

$$\begin{bmatrix} x' \\ y' \end{bmatrix} = \begin{bmatrix} \cos \theta & -\sin \theta \\ \sin \theta & \cos \theta \end{bmatrix} \begin{bmatrix} x \\ y \end{bmatrix}$$



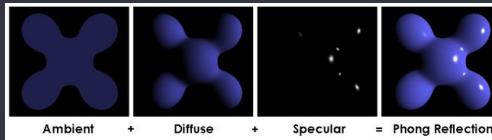
gl

experimental, rotate image, ray marching, dem



gl

experimental, rotate image, ray marching, dem



Google Earth Engine

Search places and datasets... q

ScreenToGif - Snapshot mode X

Help gennadij.donchits

Layers Map Satellite

0 00 ► Hide 0.9

Google

Code

gl

experimental, rotate image, ray marching, dem

gl-test-spheres - Earth Engine

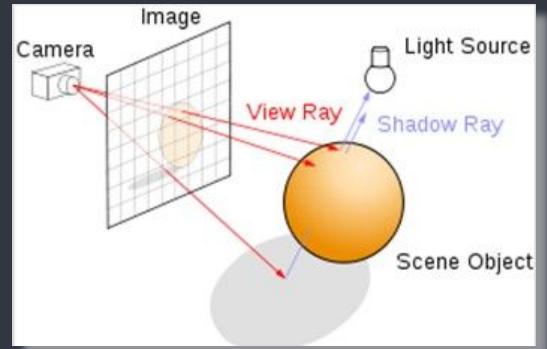
Secure | code.earthengine.google.com/0088322e2eb1336e0a0c04a9b3d162b6

Google Earth Engine Search places and datasets... Help gennadyy.donchys

gl-test-spheres *

```
i 54 // normalize ray, FOV=90
i 55 var r = gl.normalize(uv.addBands(ee.Image.constant(1).rename('z')))
i 56
i 57 // rotate r around y axis
i 58 var angle = time.multiply(Math.PI / 4)
i 59 r = gl.rotate(r, [1, 0], angle)
i 60
i 61 // origin
i 62 var o = ee.Image([0, z, z])
i 63
i 64 // render
i 65 var t = trace(o, r)
i 66
i 67 // fog function 1/(1 + 0.1 * t^2)
i 68 var fog = ee.Image.constant(1).divide(ee.Image.constant(1.0).add(t.multiply(t).multiply(0.1)))
i 69
```

Layers Map Satellite



Code

The end of part I



Icons / images in EE

Google Earth Engine ee-brabben

```
2022-04-15 icons in ee
1
2
3 // 
4 // 
5 // 
6 // 
7
8 // You can include Material icons served from https://fonts.google.com.
9 // 1. Go to https://fonts.google.com/icons
10 // 2. Find and select the icon you like
11 // 3. In lower right, right click SVG button and copy the URL
12 // 4. Paste URL as imageUrl parameter argument.
13 print(ui.Label({imageUrl: 'https://fonts.gstatic.com/s/i/materialiconsoutlined/rocket_launch/v1/24px.svg'}));
14 print(ui.Label({imageUrl: 'https://fonts.gstatic.com/s/i/materialiconsoutline/satellite_ait/v1/20px.svg'}));
15 print(ui.Label({imageUrl: 'https://fonts.gstatic.com/s/i/materialicons/location_on/v1/24px.svg'}));
16 print(ui.Label({imageUrl: 'https://fonts.gstatic.com/s/i/materialiconsround/park/v1/24px.svg'}));
17 print(ui.Label({imageUrl: 'https://fonts.gstatic.com/s/i/materialiconsoutlined/pedal_bike/v14/24px.svg'}));
18
19
20
21 // 
22 // 
23 // 
24 // 
25
26 // You can also include images served from https://www.gstatic.com. These are images that Google hosts,
27 // mostly for branding. Here, the Earth Engine logo is shown. There is not a lot you can do with the gstatic
28 // option.
29 print(ui.Label({imageUrl: 'https://www.gstatic.com/images/icons/product/earth_engine-128.png'}));
30 // Earth Engine logo
31
32
33
34 // 
35 // 
36 // 
37 // 
38
39 // 1. Make a JSON file with some data URIs (use Python's base64 package for encoding)
40 // 2. Upload JSON file to your Storage Bucket
41 // 3. Load JSON file as blob in and process to dictionary
42 // 4. Fetch image data URIs from dictionary
43 var eeDict = ee.Blob('gs://ee-blob-test/blog.json').string().decodeJSON();
44 eeDict.evaluate(function(eeDict) {
45   print(ui.Label({imageUrl: eeDict.img1}));
46 });


```

Get Link Save Run Reset Apps

Inspector Console Tasks

Use print(...) to write to this console.

Print icon: A small blue rocket ship.

Console icon: A small gear and wrench.

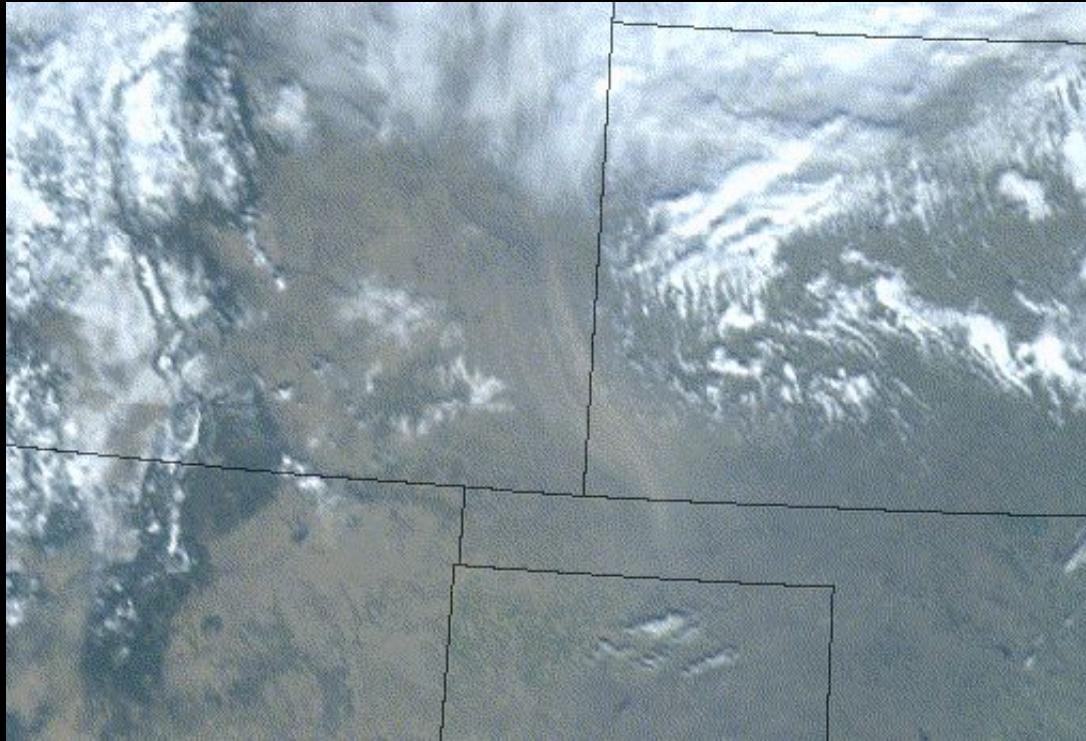
Tasks icon: A small location pin.

Map icon: A small north arrow.

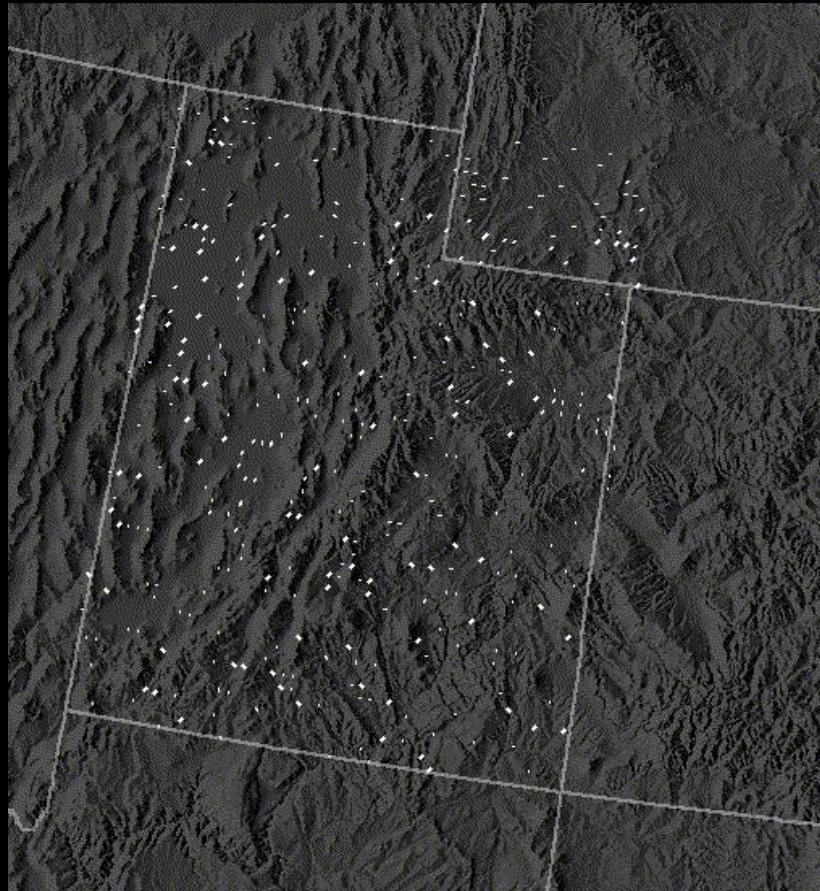
EE Logo icon: A stylized yellow pencil character with a face.

<https://code.earthengine.google.com/d2308d6a5cbb18ba9ec764370eb3084f>

Use ee.Image.translate to shift each frame



#GeoForGood22



#GeoForGood22

<https://gist.github.com/jdbcde/a6988f20f6dd8bc6d43f903c68e6e3>



#GeoForGood22

DEMO: Landsat false-color visualization & map tile export

The screenshot shows the Google Earth Engine code editor interface. The left panel displays a sidebar with various Earth Engine API categories like ee.Algorithms, ee.Array, ee.Blob, etc. The main workspace contains a script titled "Link 18aa7518e8385e8202d15ecab9980aa". The script performs several steps:

- It loads a FeatureCollection of state boundaries.
- It defines a function to scale and mask Landsat 2 SR data.
- It filters the data by state and date.
- It creates a median composite image.
- It defines visualization parameters for an RGB visualization.
- It clips the visualization to state boundaries.
- It adds the visualization layer to a map.
- Finally, it exports the map tiles to cloud storage.

The right panel shows a map of the western United States, specifically focusing on Nevada, Utah, and Colorado. The map displays a false-color visualization of the Landsat data, showing different land cover types in various colors. A legend on the right side of the map identifies the colors used in the visualization.

<https://code.earthengine.google.com/83c943331bb7aed7f6d23b593d90c827>

#GeoForGood22

Exporting Map Tiles

```
40 // Export map tiles to cloud storage for use in Earth and Google Maps.  
41 Export.map.toCloudStorage({  
42   image: imgVis,  
43   description: 'ca-false-color',  
44   bucket: 'ee-demos',  
45   fileFormat: 'auto',  
46   path: 'g4g/ca-false-color',  
47   maxZoom: 12,  
48   region: state,  
49   writePublicTiles: true,  
50   bucketCorsUris: [  
51     'https://code.earthengine.google.com',  
52     'https://*.earthengine.app',  
53     'https://earth.google.com'  
54   ]  
55 });  
--
```

https://developers.google.com/earth-engine/guides/exporting_map_tiles

Setting up a Bucket

on Google Cloud Storage

- Go to: <https://console.cloud.google.com/>
- Choose your project (dropdown in top blue bar)
- Go to: ≡ Menu >> Storage >> Cloud Storage >> Buckets
- Click "Create" or "Create Bucket" button
- Choose settings:
 - Name
 - Location
 - Storage class
 - Access control
 - Protection / versioning

Why host tiles in a GCS Bucket?

Why not Google Drive?

Name your bucket

Pick a globally unique, permanent name. [Naming guidelines](#)

Tip: Don't include any sensitive information

LABELS (OPTIONAL)

[CONTINUE](#)

Choose where to store your data

This permanent choice defines the geographic placement of your data and affects cost, performance, and availability. [Learn more](#)

Location type

 Multi-region

Highest availability across largest area

 Dual-region

High availability and low latency across 2 regions

 Region

Lowest latency within a single region

[CONTINUE](#)

Choose a default storage class for your data

A storage class sets costs for storage, retrieval, and operations. Pick a default storage class based on how long you plan to store your data and how often it will be accessed. [Learn more](#)

 Standard [?](#)

Best for short-term storage and frequently accessed data

 Nearline

Best for backups and data accessed less than once a month

 Coldline

Best for disaster recovery and data accessed less than once a quarter

 Archive

Best for long-term digital preservation of data accessed less than once a year

[CONTINUE](#)

Choose how to control access to objects

Prevent public access

Restrict data from being publicly accessible via the internet. Will prevent this bucket from being used for web hosting. [Learn more](#)

 Enforce public access prevention on this bucket

Access control

 Uniform

Ensure uniform access to all objects in the bucket by using only bucket-level permissions (IAM). This option becomes permanent after 90 days. [Learn more](#)

 Fine-grained

Specify access to individual objects by using object-level permissions (ACLs) in addition to your bucket-level permissions (IAM). [Learn more](#)

[CONTINUE](#)

Choose how to protect object data

Your data is always protected with Cloud Storage but you can also choose from these additional data protection options to prevent data loss. Note that object versioning and retention policies cannot be used together.

Protection tools

 None Object versioning (best for data recovery)

For restoring deleted or overwritten objects. To minimize the cost of storing versions, we recommend limiting the number of noncurrent versions per object and scheduling them to expire after a number of days. [Learn more](#)

 Retention policy (best for compliance)

For preventing the deletion or modification of the bucket's objects for a specified minimum duration of time after being uploaded. [Learn more](#)

DATA ENCRYPTION

[CREATE](#)[CANCEL](#)

Public access will be prevented

This bucket is set to prevent exposure of its data on the public internet.

Keep this setting enabled unless you have a use case that requires public access (such as static website hosting). You can change it now or later. [Learn more](#)

 Enforce public access prevention on this bucket Don't show this message again[CANCEL](#) [CONFIRM](#)

Map Tile Export Task results - with output urls & links

The screenshot shows the Google Earth Engine Tasks interface. At the top, there are three tabs: Inspector, Console, and Tasks. The Tasks tab is selected, showing a list of tasks. The first task in the list is "mapTilesForEarthWeb". Below the task list, detailed information about the task is provided:

- ID: C7VOJU7LDROPR32KRKIYJZYL
- Phase: Completed
- Runtime: 6m (started 2022-09-23 00:20:06 -0700)
- Attempted 1 time
- Batch compute usage: 464.2327 EECU-seconds

Under the "Output URLs" section, two URLs are listed:

- Google Earth Tiles URL:
https://storage.googleapis.com/test20220923v05/mapTiles
ForEarthWeb/\${level}/\${x}/\${y}
- Google Maps Tiles URL:
https://storage.googleapis.com/test20220923v05/mapTiles
ForEarthWeb/{z}/{x}/{y}

At the bottom of the interface are four buttons: View task, Source Script, Open in GCS, and Open in Google Earth.

New Guide: https://developers.google.com/earth-engine/guides/exporting_map_tiles

#GeoForGood22

Map Tile Export - Task output to GCS Bucket

[https://console.cloud.google.com/storage
/browser/ee-demos/g4g/ca-false-color](https://console.cloud.google.com/storage/browser/ee-demos/g4g/ca-false-color)

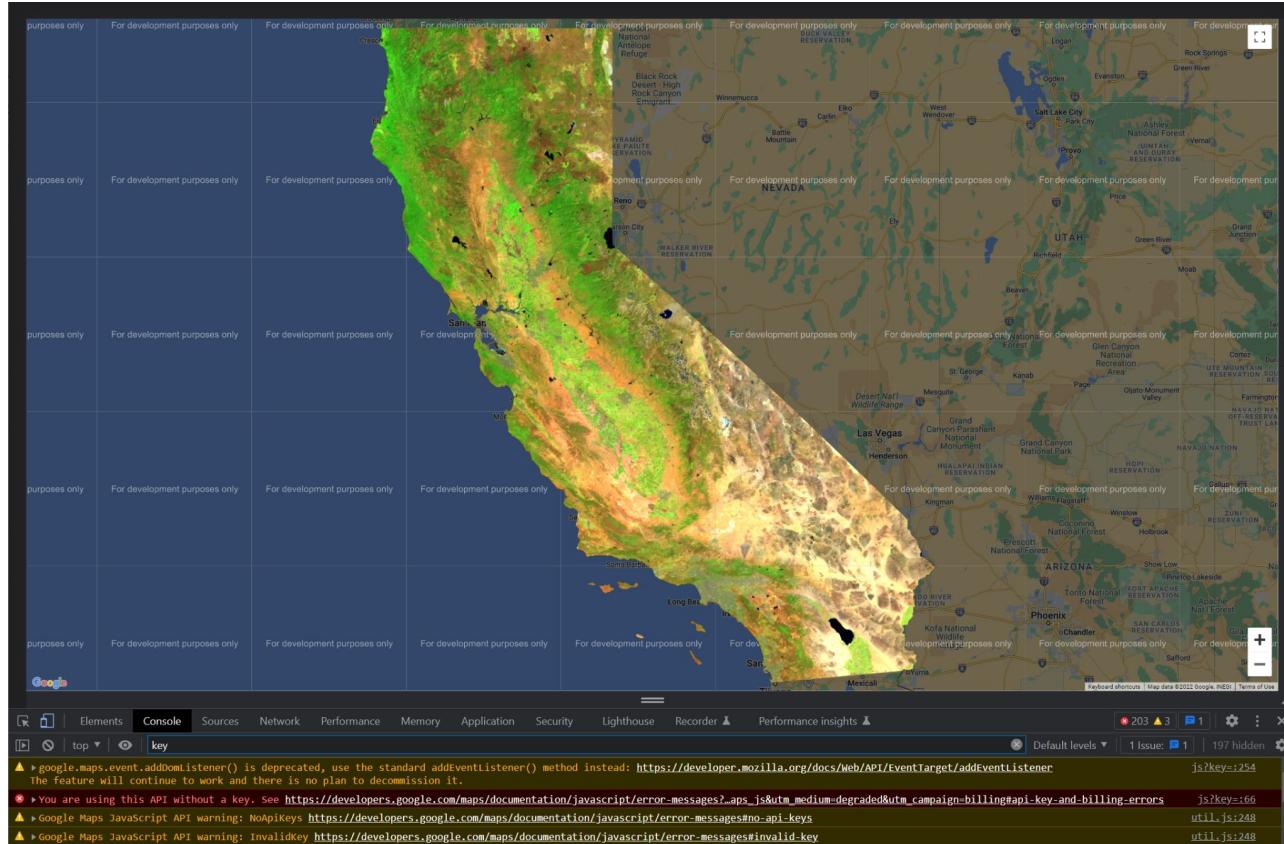
Note:

- Tile folder structure /z/x/y.ext
- File types: PNG/JPG/none
- Maps preview: [index.html](#)
- Earth preview: [earth.html](#)
- Maps API Key
- ACLs / permissions
- JS Access/CORS settings (at the bucket level)
- Check or modify JS Access/CORS
gsutil cors get gs://BUCKET_NAME
gsutil cors set CORS_CONFIG_FILE
gs://BUCKET_NAME
- Delete tiles: Console or Cloud shell

Bucket details						
ee-demos		Location	Storage class	Public access	Protection	
us (multiple regions in United States)		Standard	Subject to object ACLs	None		
OBJECTS	CONFIGURATION	PERMISSIONS	PROTECTION	LIFECYCLE	OBSERVABILITY	
Buckets > ee-demos > g4g > ca-false-color						
UPLOAD FILES	UPLOAD FOLDER	CREATE FOLDER	TRANSFER DATA ▾	MANAGE HOLDS	DOWNLOAD	DELETE
Filter by name prefix only	Filter	Filter objects and folders				
<input type="checkbox"/>	Name	Size	Type	Created	Storage class	Last modified
<input type="checkbox"/>	0/	—	Folder	—	—	—
<input type="checkbox"/>	1/	—	Folder	—	—	—
<input type="checkbox"/>	10/	—	Folder	—	—	—
<input type="checkbox"/>	11/	—	Folder	—	—	—
<input type="checkbox"/>	12/	—	Folder	—	—	—
<input type="checkbox"/>	2/	—	Folder	—	—	—
<input type="checkbox"/>	3/	—	Folder	—	—	—
<input type="checkbox"/>	4/	—	Folder	—	—	—
<input type="checkbox"/>	5/	—	Folder	—	—	—
<input type="checkbox"/>	6/	—	Folder	—	—	—
<input type="checkbox"/>	7/	—	Folder	—	—	—
<input type="checkbox"/>	8/	—	Folder	—	—	—
<input type="checkbox"/>	9/	—	Folder	—	—	—
<input type="checkbox"/>	earth.html	2.4 KB	text/html; charset=utf-8	Sep 27, 20...	Standard	Sep 27, 20... ▲ Public to internet Copy URL
<input type="checkbox"/>	index.html	1.5 KB	text/html; charset=utf-8	Sep 27, 20...	Standard	Sep 27, 20... ▲ Public to internet Copy URL

Google Maps preview

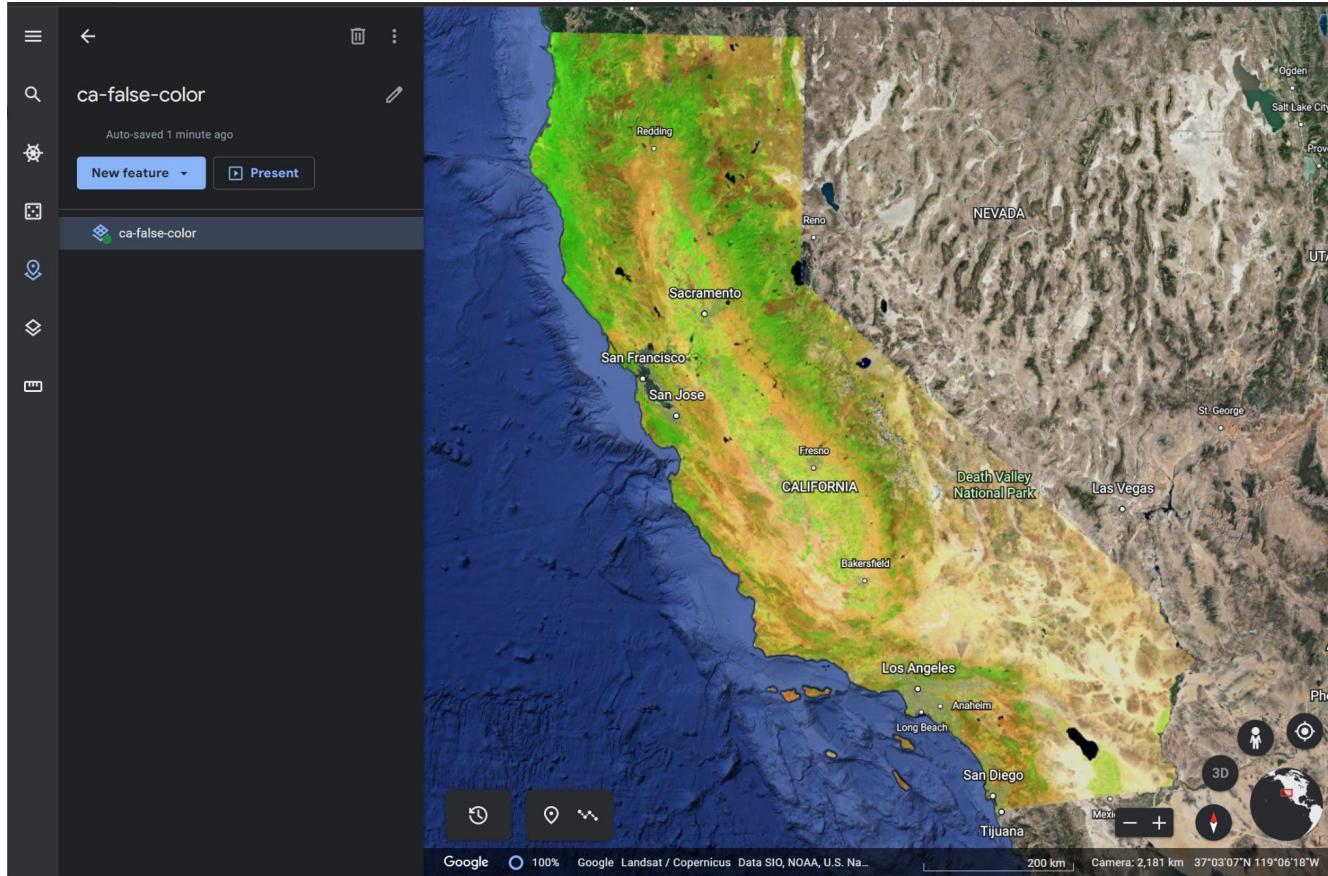
<https://storage.googleapis.com/ee-demos/g4g/ca-false-color/index.html>



#GeoForGood22

Google Earth preview

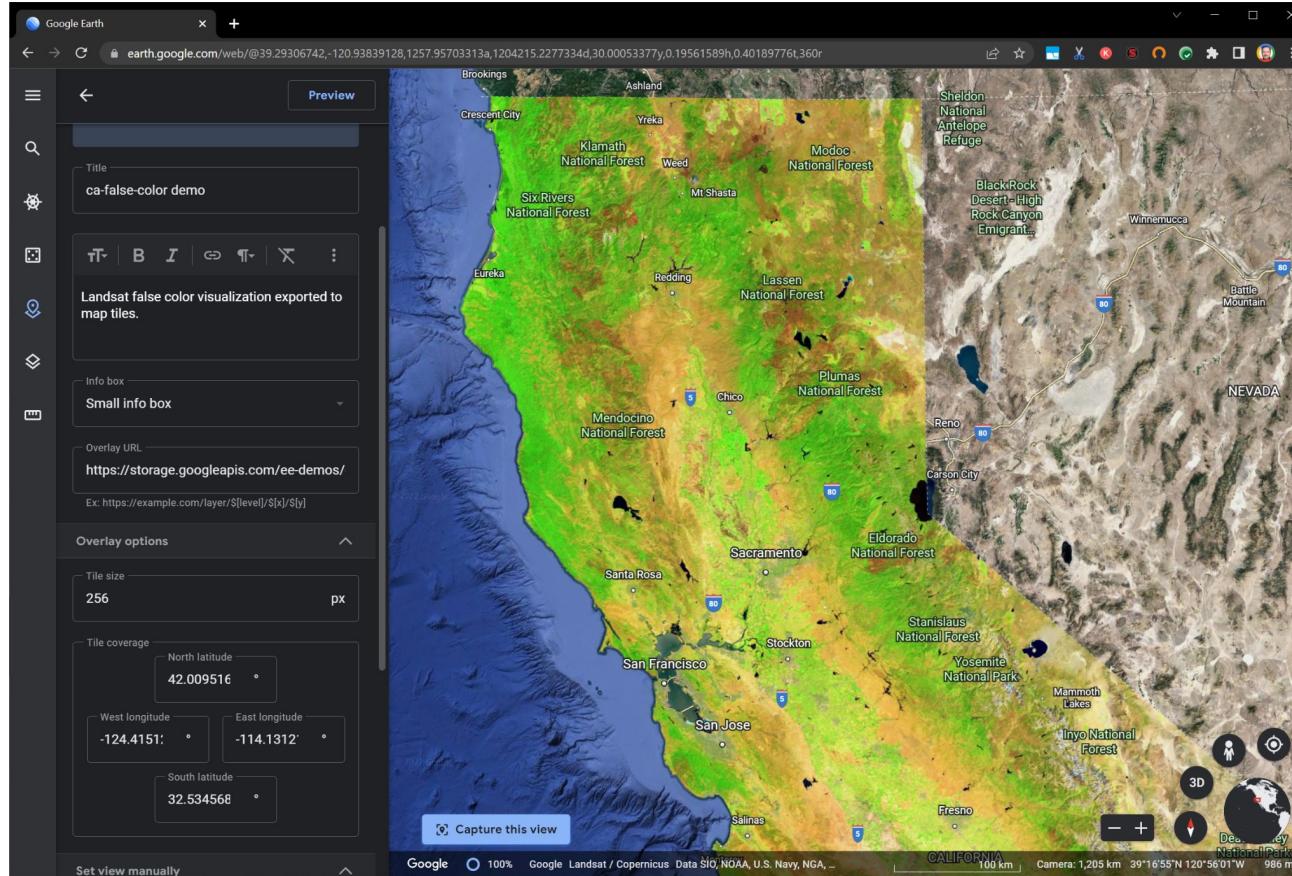
<https://storage.googleapis.com/ee-demos/g4g/ca-false-color/earth.html>



#GeoForGood22

Create Tile Layer in Google Earth web

Earth Project: https://earth.google.com/earth/d/1TvDkzIMON_7jdeSxXTp9pP77JuYK_lcy



Tile URL for Earth:

[https://storage.googleapis.com/ee-demos/g4g/ca-false-color/\\$\[level\]/\\$\[x\]/\\$\[y\]](https://storage.googleapis.com/ee-demos/g4g/ca-false-color/$[level]/$[x]/$[y])

#GeoForGood22

Export vector data to KML

Historical Wildfire Boundaries ([MTBS dataset](#))

Attempts to reduce file size:

- Filter to CA fires, export everything:

<https://code.earthengine.google.com/b3abab18a9e63348b2619c9d55e045ce>

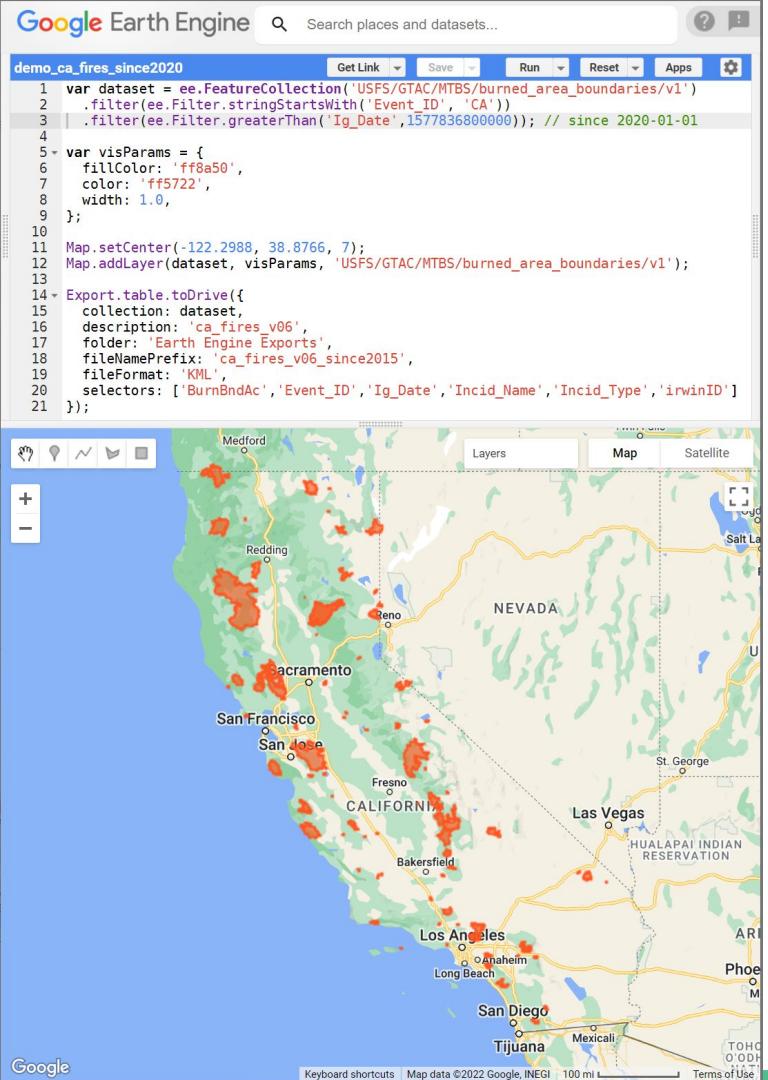
- Filter to CA since 2020, limit included attributes:

<https://code.earthengine.google.com/ca88bba37490737a489304f1b3cef695>

- Filter to CA since 2020, simplify geometries (50m limit):

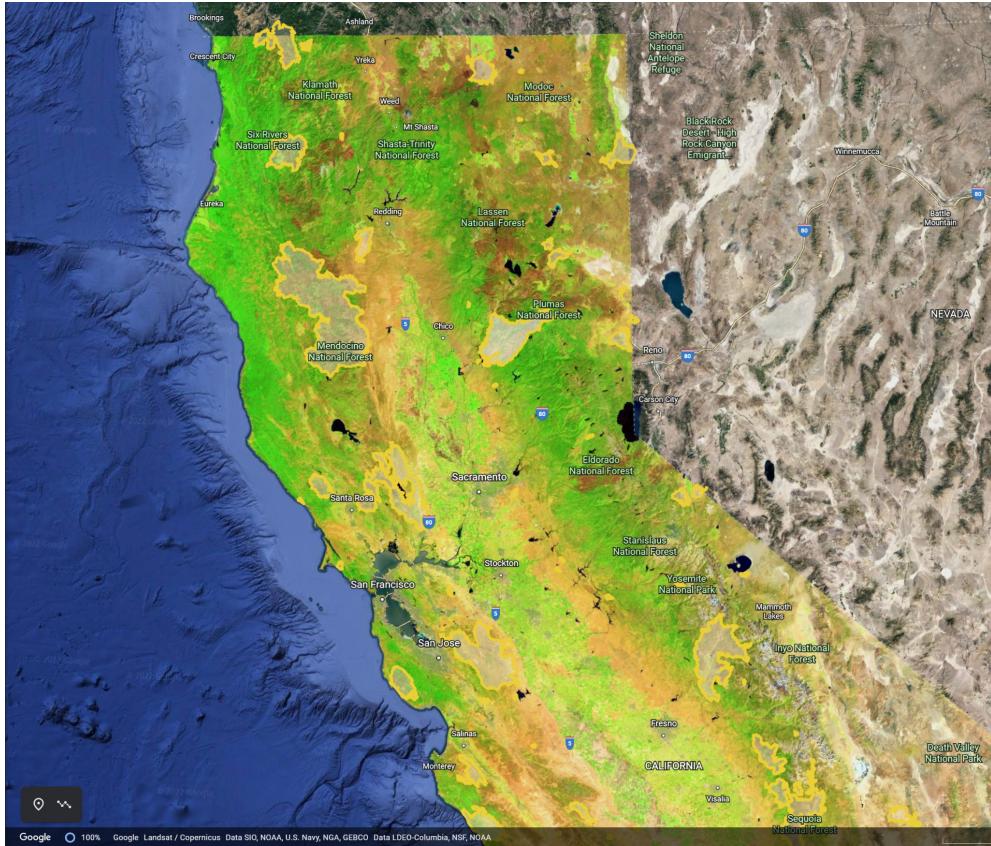
<https://code.earthengine.google.com/cc704b98c1402aaebfd381f08454e91>

#GeoForGood22



Raster + Vector ... Map Tiles + KML !!

Difficult to combine into one project, but can overlay as two projects:

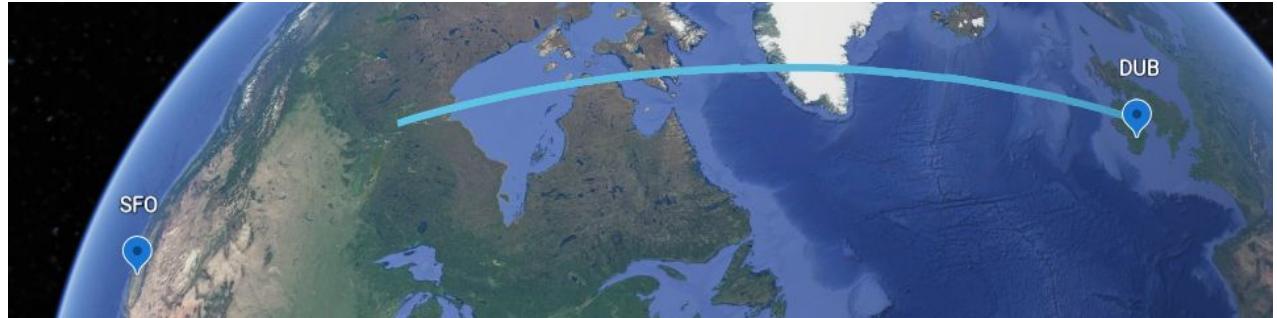


#GeoForGood22

Back down to Earth...

Overlays in Earth Web

- Controlling Earth overlays from panel HTML
- Animating Earth overlays with Javascript
- Working with IDs in Earth cloud projects



Controlling Earth overlays from panel HTML

← Preview

HTML Auto-saved seconds ago

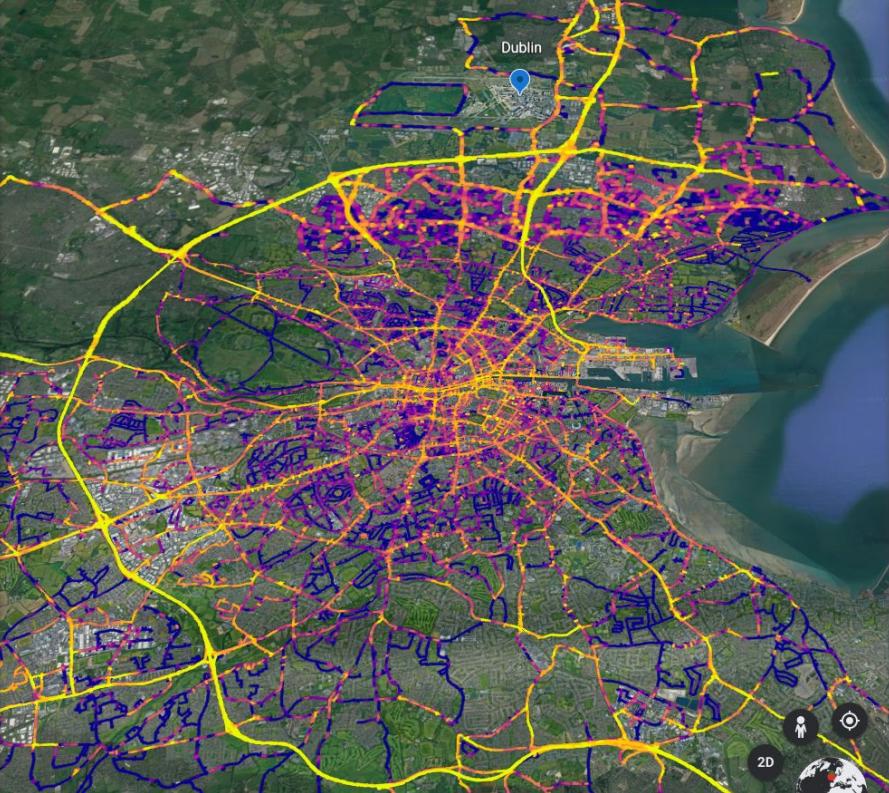
Title Dublin

This area will allow you to edit the content as HTML.

```
<br>

<a href="#overlayId;show">#overlayId;show</a><br>
<a href="#overlayId;hide">#overlayId;hide</a><br>
<br>
<a href="#placemarkId;teleport">#placemarkId;teleport</a><br>
<a href="#placemarkId;">
```

Switch to template



X



Panel title

Lore ipsum dolor sit amet, consectetur adipisicing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur.

#overlayId:show
#overlayId:hide

#placemarkId:teleport
#placemarkId:flyto
#placemarkId:contextual
#placemarkId:balloon
#placemarkId:balloonFlyto
#placemarkId:flytoOrbit

#GeoForGood22

Animating Earth overlays with Javascript

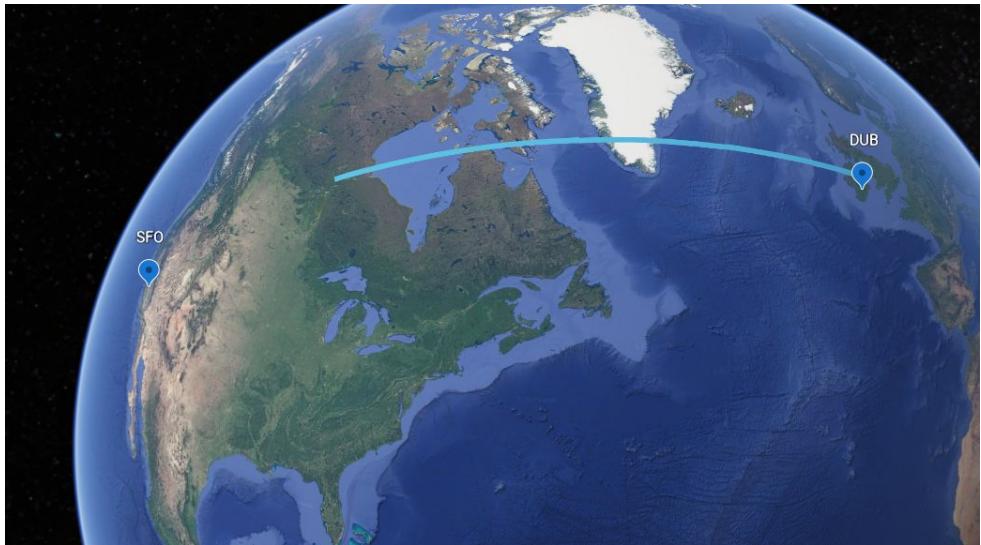
```
<a href="" id="clicker"></a>
<script>
const ids=["step0","step1","step2","step3","step4","step5","step6"

for (var i=0; i < ids.length; i++) {
  var id=ids[i];
  var hide="#"+id+";hide";
  var show="#"+id+";show";
  clicker(hide,0);
  clicker(show,1000+i*220);
}
clicker("#sfo;flyto",1500);

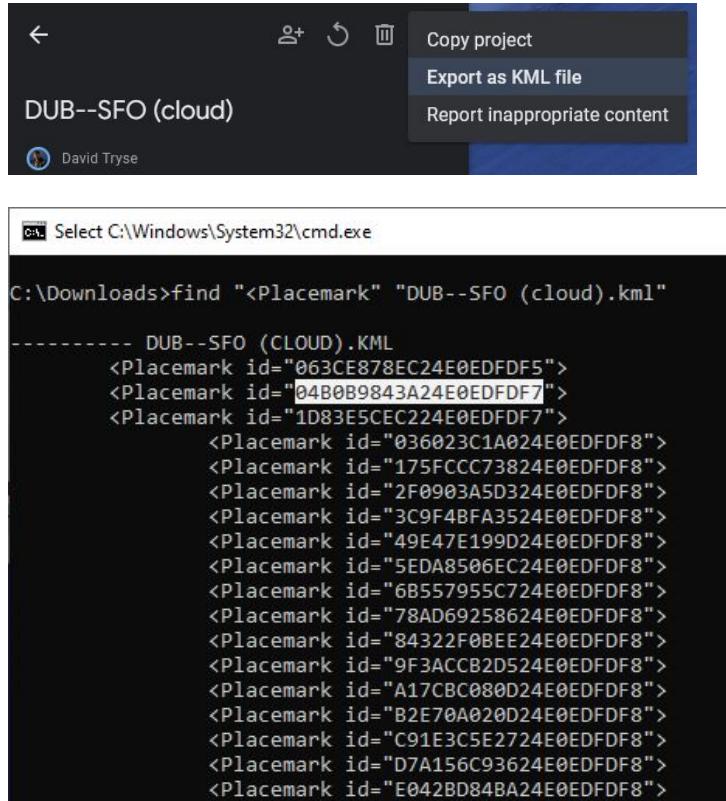
function clicker(href,delay) {
  setTimeout(function() {
    var clickElem=document.getElementById("clicker");
    clickElem.href=href;
    clickElem.click();
  }, delay);
}

</script>

<Folder>
<name>flight-path</name>
<Placemark id="step0"><visibility>0</visibility><styleUrl>#line</styleUrl><LineString><coordinates>-6.249782,53.426312 -6.416065,53.515928</coordinates></LineString></Placemark>
<Placemark id="step1"><visibility>0</visibility><styleUrl>#line</styleUrl><LineString><coordinates>-6.416065,53.515928 -6.607492,53.617887</coordinates></LineString></Placemark>
<Placemark id="step2"><visibility>0</visibility><styleUrl>#line</styleUrl><LineString><coordinates>-6.607492,53.617887 -7.14056,53.8989</coordinates></LineString></Placemark>
<Placemark id="step3"><visibility>0</visibility><styleUrl>#line</styleUrl><LineString><coordinates>-7.14056,53.8989 -7.719875,54.196597</coordinates></LineString></Placemark>
<Placemark id="step4"><visibility>0</visibility><styleUrl>#line</styleUrl><LineString><coordinates>-7.719875,54.196597 -8.856497,54.765221</coordinates></LineString></Placemark>
<Placemark id="step5"><visibility>0</visibility><styleUrl>#line</styleUrl><LineString><coordinates>-8.856497,54.765221 -9.839295,55.230153</coordinates></LineString></Placemark>
```



Working with IDs in Earth cloud projects



```
<br>
<a href="#357392746424E0EDFDF7;show">#no2;show</a><br>
<a href="#357392746424E0EDFDF7;hide">#no2;hide</a><br>

<a href="" id="clicker"></a>
<script>
const ids=[ "036023C1A024E0EDFDF8", "175FCCC73824E0EDFDF8", "2F0903A5D3

for (var i=0; i < ids.length; i++) {
    var id=ids[i];
    var hide="#"+id+";hide";
    var show="#"+id+";show";
    clicker(hide,0);
    clicker(show,1000+i*220);
}
clicker("#04B0B9843A24E0EDFDF7;flyto",1500);

function clicker(href,delay) {
    setTimeout(function() {
        var clickElem=document.getElementById("clicker");
        clickElem.href=href;
        clickElem.click();
    }, delay);
}

</script>
```

Thank You!

bit.ly/g4g22-hackeaerth

Questions?

- How do I... ? Is it possible to... ?
- What do you want to know more about?
- What are you inspired by?