

# Zpracování družicových dat v cloudu



Mgr. Daniel Paluba  
([palubad@natur.cuni.cz](mailto:palubad@natur.cuni.cz))



UNIVERZITA KARLOVA  
Přírodovědecká fakulta  
Katedra aplikované geoinformatiky a kartografie

# Obsah dnešního workshopu

## **Seznámení s Google Earth Engine + ukázky**

- Představení Google Earth Engine (GEE)
- Data, datové typy a funkce v GEE
- Ukázky aplikací postavené na GEE
- GEE Timelapse

## **Praktická část**

- Představení GEE Code Editoru
- Import a zobrazení dat
- Vytvoření bezoblačné mozaiky světa ze všech snímků Sentinel-2 v roce 2020
- Časové řady





Earth

Google Earth Engine combines a multi-petabyte catalog of satellite imagery and geospatial datasets with planetary-scale analysis capabilities and makes it available for scientists, researchers, and developers to detect changes, map trends, and quantify differences on the Earth's surface



# Earth

- + Zdarma pro nekomerční užití
- + Velké množství předdefinovaných funkcí
- + Široký výběr dat
- + Obrovská uživatelská komunita
- + Možnost vytvořit si vlastní aplikace
- + Možnost skriptování i v mobilu
- Potřeba znalosti programování v JS nebo Pythonu
- „Closed software“; funkce se mohou měnit časem – potřeba si hlídat aktualizace
- Chybějící záruka kontinuity platformy

The screenshot displays the Google Earth Engine web interface. The top navigation bar includes 'Scripts', 'Docs', and 'Assets'. The main workspace is divided into three panels:

- Scripts Panel:** Shows a script named 'LC-SLIAC' with the following code:

```
378 // Add corrected values to the Sentinel-1 ImageCollection
379 var addCorrectedValues = function(img) {
380
381   var VV = img.select('VV'),
382       VH = img.select('VH'),
383       VVscale = ee.Image(ee.Number(img.get('VVscale'))),
384       VHscale = ee.Image(ee.Number(img.get('VHscale'))),
385       angleDiff = (img.select('LIA').subtract(referenceAngle)),
386       radarAngle = img.select('angle'),
387       LIA = img.select('LIA');
388
389   var corrected_VV = VV.subtract((VVscale.multiply(angleDiff))
390     .rename('Corrected_VV'));
391
392   var corrected_VH = VH.subtract((VHscale.multiply(angleDiff))
393     .rename('Corrected_VH'));
394
395   return img.addBands([corrected_VH, corrected_VV]);
396
397 };
398
399 // Add corrected bands to the Image Collection
400 var correctedValues = ImgCollWithRegression.map(addCorrectedValues);
401
402 return correctedValues;
403
404 };
405 // export the function
406 exports.LC_SLIAC = LC_SLIAC;
```
- Inspector Panel:** Shows the output of the script, including a time-series plot titled 'Time-series of corrected and uncorrected data'. The plot shows two data series: 'VH' (blue) and 'VV' (red) over time from September 2018 to May 2019. The y-axis is labeled 'Band Values' and ranges from -30 to 0.
- Console Panel:** Displays the message: 'Use print(...) to write to this console.'

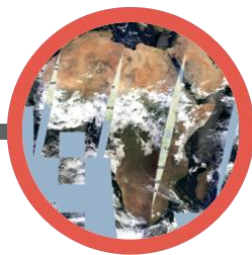
At the bottom of the interface, a satellite map is visible, showing a forested area with a road. A smartphone is overlaid on the left side of the map, displaying the same interface on a mobile device.

# Data v GEE

<https://developers.google.com/earth-engine/datasets/>



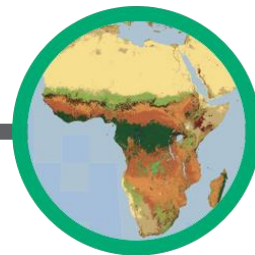
**Landsat & Sentinel-2**  
10-30m, 14-day



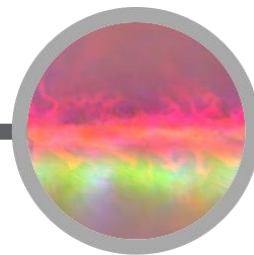
**MODIS**  
250m daily



**Sentinel-1**  
Radar



**Terrain & Land Cover**



**Weather & Climate**  
NOAA NCEP, OMI, ...

... and you can upload your own imagery

> 700 public datasets

> 5 million images

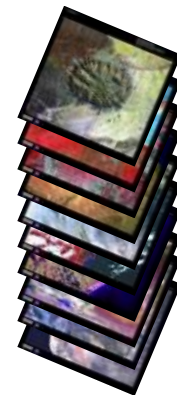
> 4000 new images every day

~ 35 petabytes of data



# Datové typy

- **ee.Image** – raster, 1 nebo více pásem; každé pásmo má vlastní jméno, měřítko (scale), projekci a masku + metadata jako množinu atributů (properties).
- **ee.ImageCollection** – kolekce souvisejících snímků (ee.Image) se stejnou charakteristikou a vlastnostmi. Vlastní ID.
- **ee.Feature** – vektor, jako objekt GeoJSON. Dvě vlastnosti: 1) Geometry - geometrie (případně NULL) a 2) Property - metadata, které obsahuje jednotlivé atributy.
- **ee.Geometry** - Point, LineString (a list of points), LinearRing (a closed LineString), and Polygon (a list of LinearRings where the first is a shell and subsequent rings are holes). MultiPoint, MultiLineString, and MultiPolygon. MultiGeometry (GeoJSON GeometryCollection).
- **ee.FeatureCollection** – kolekce souvisejících objektů (ee.Feature).
- **ee.Array** - pole s 1D vektory, 2D matice, 3D cubes nebo vícedimenzionální prostory
- **ee.Number, ee.String, ee.Date, ee.List, ee.Dictionary,...**

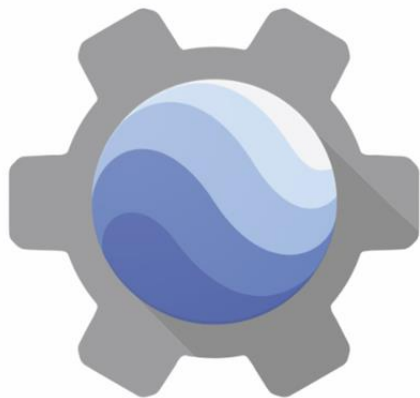


# Geoprostorové funkce

- **Filter** - by bounds, within distance, date, day-of-year, metadata ...
- **Reducer** - způsob agregace dat v čase, prostoru, pásmech, polích a dalších datových strukturách, např. mean, max, min, lineární regrese, histogram, zonal statistics ....
- **Join** - kombinování prvků z různých kolekcí (např. ImageCollection nebo FeatureCollection) na základě podmínky určené ee.Filter, např. simple, inner, outer, inverted, spatial join ...
- **Základní GIS funkce** - clip, buffer, intersect, union, dissolve, ...
- **Chart** - integrace Google Charts. Column, pie chart, scatter plot, histogram, time-series, ...
- **Export** - export obrázků, mapových dlaždic, tabulek a videí
- **Machine Learning** - Řízené a neřízené per-pixel klasifikace, OBIA. Např. CART, random forests, bayes, SVM, k-means, cobweb
- **Pokročilé funkce** - Fmask, Change Detection algoritmy, ...

více než 800 předdefinovaných funkcí... a stále se rozšiřující možnosti

# Jak používat Google Earth Engine?



## JavaScript

Interactive JavaScript using the Code Editor, the open source JavaScript library in Node.js (learn more about Earth Engine in Node.js), or Earth Engine Apps.



## Python

The open source Python library running in Colab, your Python environment, or App Engine (learn more about Earth Engine powered App Engine apps).



## REST

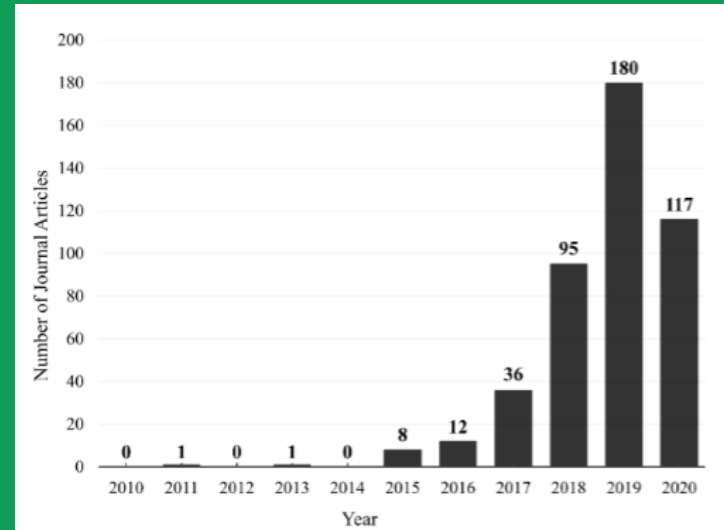
Authenticated HTTP requests (learn more about the Earth Engine REST API). The REST API contains new and advanced features that may not be suitable for all users. If you are new to Earth Engine, please get started with the JavaScript guide.

Webové integrované vývojové prostředí (IDE) – využívá EE API

Návody: <https://developers.google.com/earth-engine/guides>



# Aplikace postavené nad Google Earth Engine



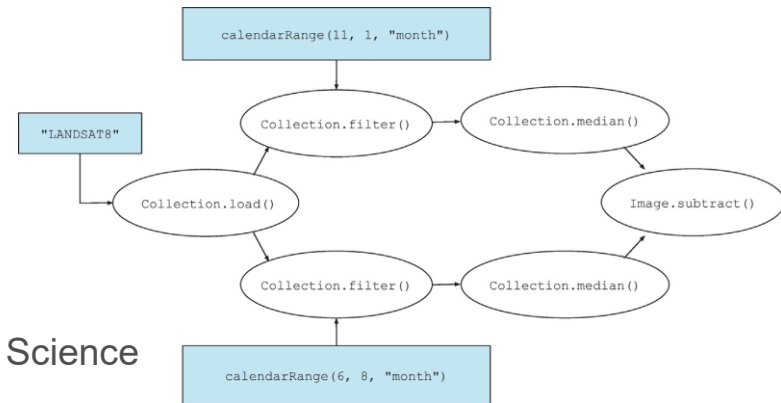
*Autor: Amani et al. 2020 (květen 2020)*

## Google Earth Engine: Planetary-scale geospatial analysis for everyone

Noel Gorelick <sup>a,\*</sup>, Matt Hancher <sup>b</sup>, Mike Dixon <sup>b</sup>, Simon Ilyushchenko <sup>b</sup>, David Thau <sup>b</sup>, Rebecca Moore <sup>b</sup>

<https://www.sciencedirect.com/science/article/pii/S0034425717302900>

```
collection = ee.ImageCollection("LANDSAT8")
winter = collection.filter(ee.Filter.calendarRange(11, 1, "month"))
summer = collection.filter(ee.Filter.calendarRange(6, 8, "month"))
diff = summer.median().subtract(winter.median())
```

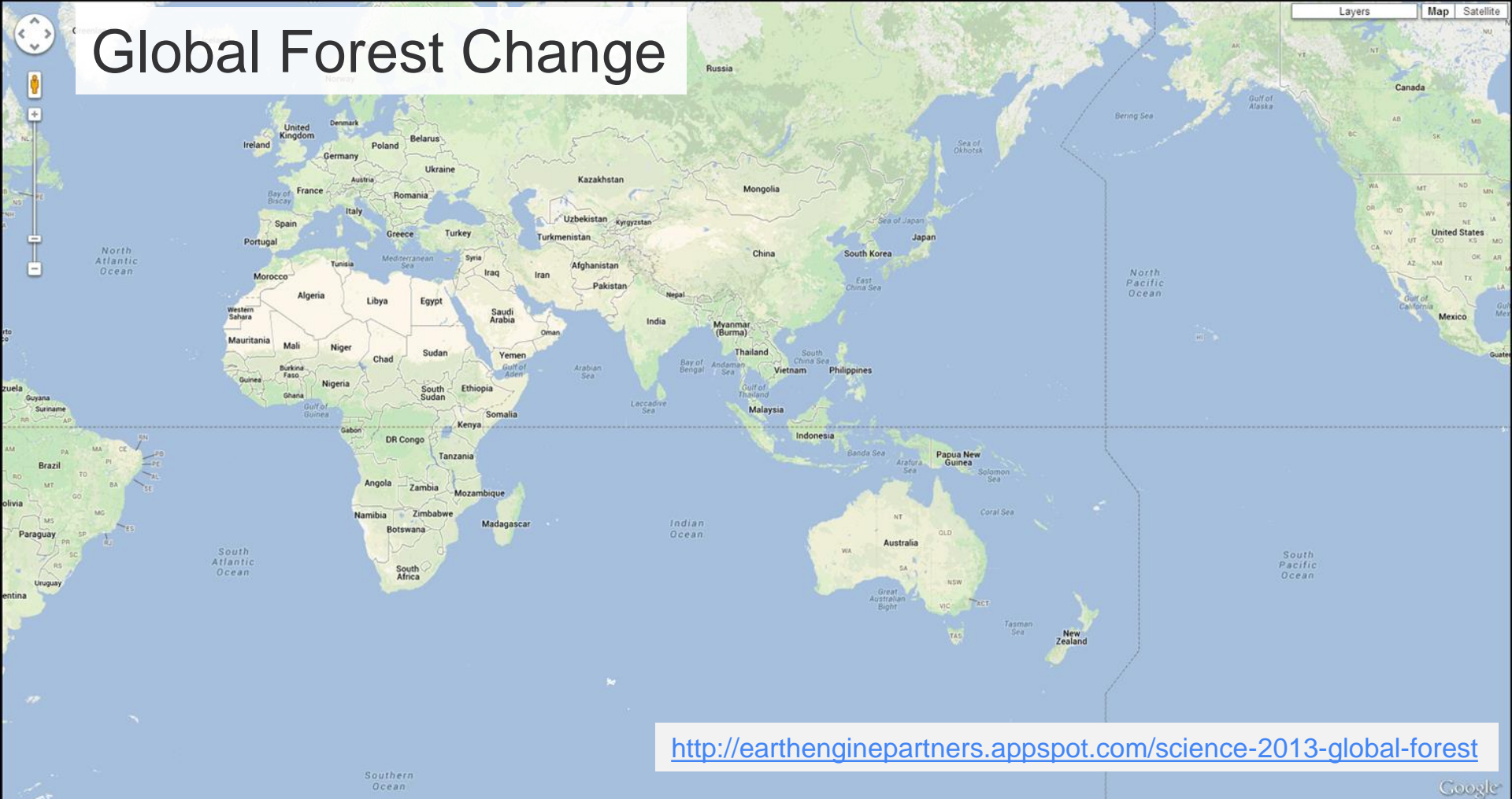


Počet citací k 09.11.2021: **2466**

16. nejvíce citovaný článek všech čas v oboru DPZ podle Web of Science

klíčová slova "Remote sensing" OR "Earth observation"

# Global Forest Change



<http://earthenginepartners.appspot.com/science-2013-global-forest>

Google

**Forest Loss in Riau, Indonesia, 2000-2012.**

Source: Hansen, Potapov, Moore, Hancher, et al. (Science, 2013).

Powered by Google Earth Engine

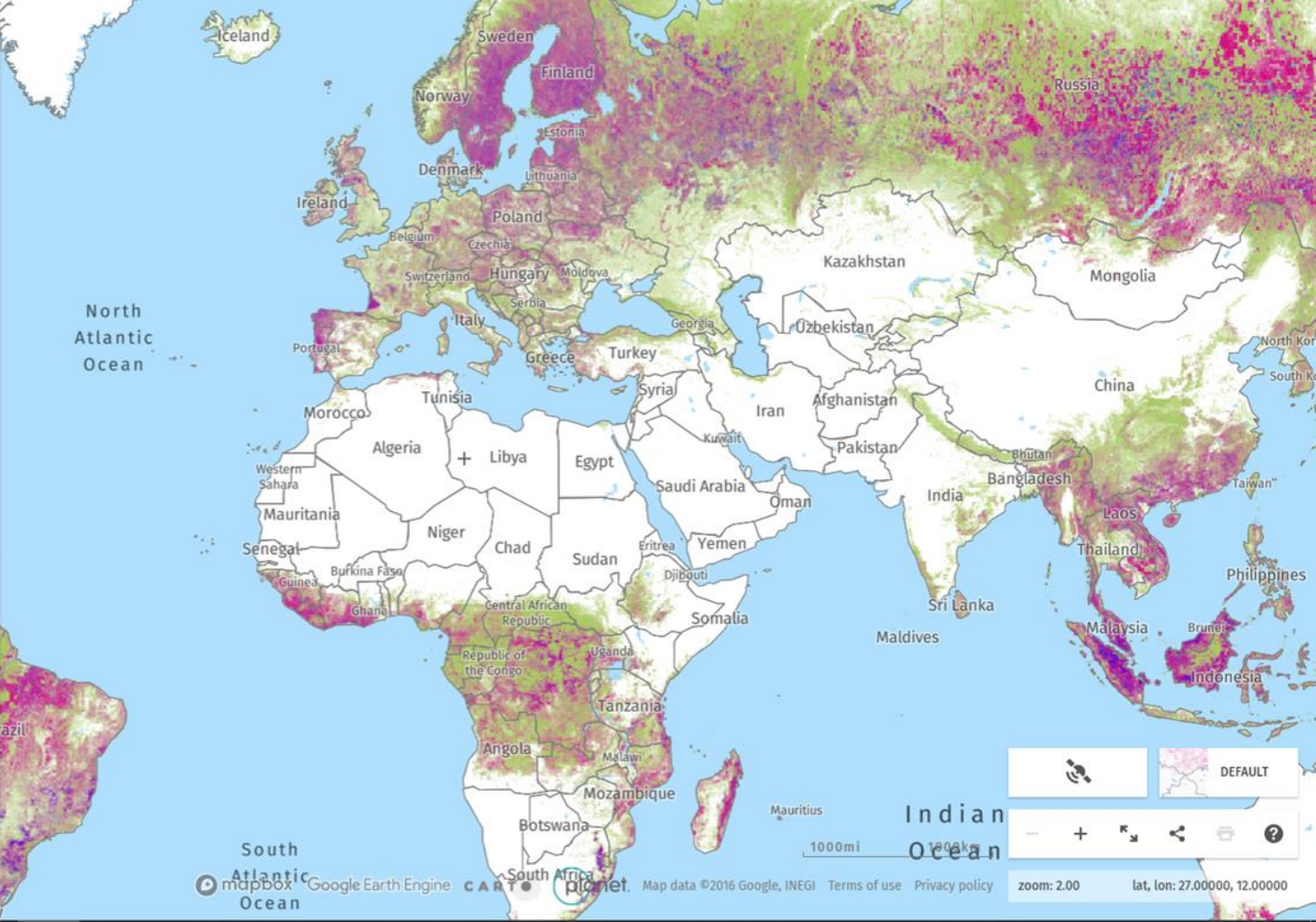
**LEGEND** **ANALYSIS**

Tree cover gain - 2001-2012  
● Tree cover gain

Tree cover loss - 2001-2019  
● Tree cover loss  
Tree cover loss is not always deforestation.  
Displaying Tree cover loss with > 30% canopy density

2001 2004 2007 2010 2013 2016 2019

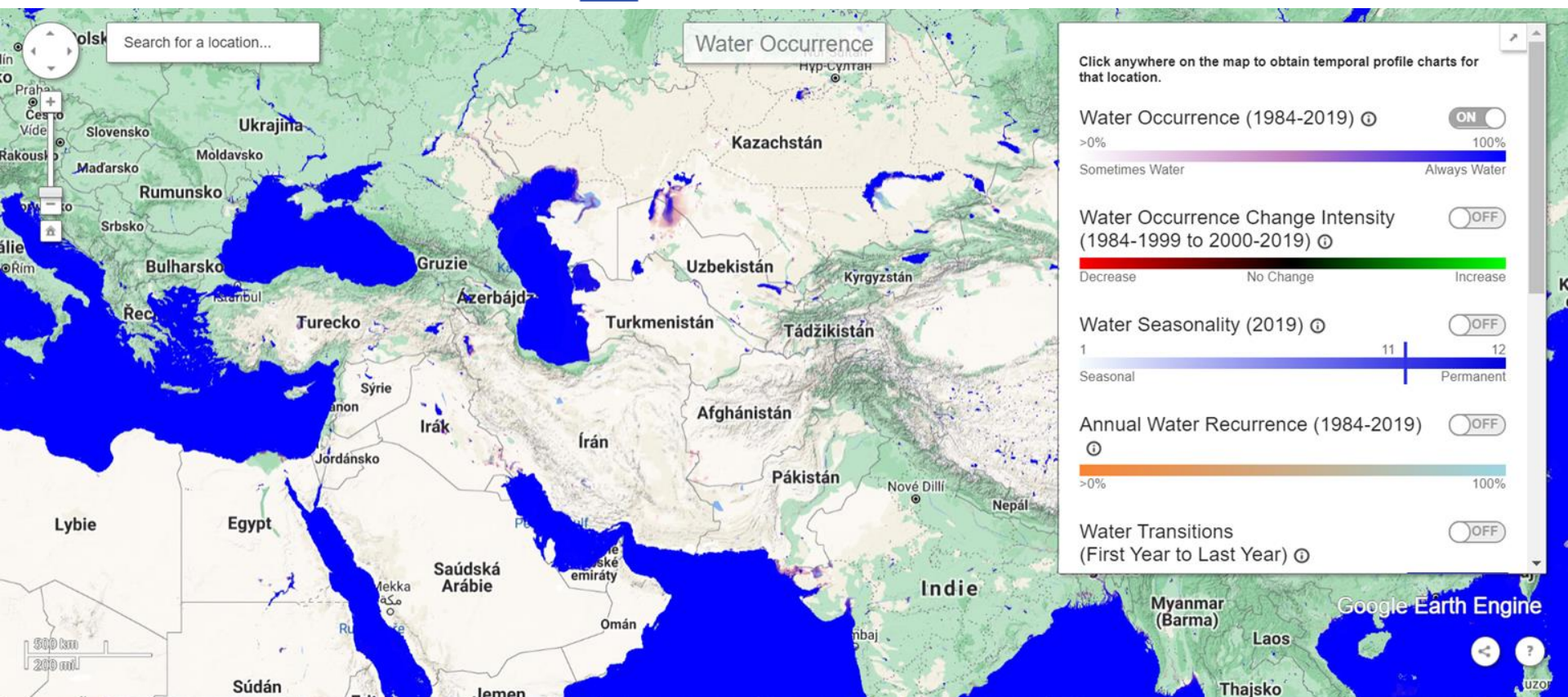
Tree cover - 2010  
● Tree cover  
Displaying Tree cover with > 30% canopy density  
Displaying Tree cover for 2010



mapbox Google Earth Engine CARTO planet. Map data ©2016 Google, INEGI Terms of use Privacy policy

zoom: 2.00 lat, lon: 27.00000, 12.00000





<https://global-surface-water.appspot.com/map>





## Tamaulipas pygmy owl

*Glaucidium sanchezi*

Learn more

Update

### Habitat Associations ON OFF

Elevation  ON  OFF

900 - 2100 meters



Tree cover:  ON  OFF

75 - 100%



Landcover  ON  OFF

Woodlands

Forests  Woody Savannas

Shrublands

Open  Closed

Herbaceous

Savannas  Grasslands

Cultivated

Cropland  Cropland Mosaics

Barren Urban

Barren  Urban

Water

Wetlands  Water Bodies

### Geographic distribution

Range size

4,509 km<sup>2</sup>

55,604 km<sup>2</sup>

Suitable

Total

Map improvement

84 %  
(81 % - 85 %)

73

Validation points

Records outside suitable range

Records in suitable range

Suitable range

Strict Parks

All Parks



### Maximum protected area coverage within suitable habitat.

	Any size	Larger than 10 km <sup>2</sup>
Strict Parks	1	1
All parks	7	3
All parks area	1,145 km <sup>2</sup>	1,136 km <sup>2</sup>
Target area	3,400.6 km <sup>2</sup>	
Target realized	33.7%	

Minimum reserve size:









EARTHMAP

The power of Google Earth  
Engine without coding.

A user friendly tool for complex  
land monitoring

 Sign in with Google

 Sign in with email

 Continue as guest

By continuing, you are indicating that you  
accept our [Terms of Service](#) and [Privacy Policy](#).

# Earth Engine Apps

Dynamic, publicly accessible user interfaces for Earth Engine analyses.

[Get started.](#)

## Curated Applications



<https://www.earthengine.app/>





# TIMELAPSE

Watch the world change over the course of nearly three decades of satellite photography

Pictured: The megacity of Dubai grows in the desert, from 1984 to today

<https://earthengine.google.com/timelapse/>

Google Earth Timelapse  
Chongqing, China  
1984



# Příklady z ČR

- Mostecko – vývoj těžby:

<https://earthengine.google.com/timelapse#v=50.52771,13.64202,9.992,latLng&t=0.03&ps=100&bt=19840101&et=20181231&startDwell=0&endDwell=0>

- Kůrovcová kalamita na Šumavě:

<https://earthengine.google.com/timelapse/#v=49.01831,13.21195,9.201,latLng&t=0&ps=50&bt=19840101&et=20181231&startDwell=0&endDwell=0>

- Obnova lesa:

<https://earthengine.google.com/timelapse#v=50.49258,13.08431,10,latLng&t=3.4&ps=50&bt=19840101&et=20181231&startDwell=0&endDwell=0>

## Praktická část

- Představení GEE Code Editoru
- Import a zobrazení dat
- Vytvoření bezoblačné mozaiky světa ze všech snímků Sentinel-2 v roce 2020
- Časové řady



# Let's Code!

[https://code.earthengine.google.com/?accept\\_repo=users/danielp/CUF2021](https://code.earthengine.google.com/?accept_repo=users/danielp/CUF2021)

## ▼ users/danielp/CUF2021

- 01. Filter an image collection
- 02. Cloudless composite of the world
- 03. Calculate NDVI and create a Time Series Analysis
- 04. Apply a computation on an image

# The Earth Engine Code Editor

The screenshot shows the Google Earth Engine Code Editor interface. The top navigation bar includes 'API Docs', 'Your Data', 'Search', 'Your Code', and 'Data Inspector'. The main workspace is divided into several panels: 'Scripts' (containing 'Your Scripts & Example Scripts'), 'Code Editor' (containing 'Your Code'), 'Inspector' (containing 'Data Inspector'), 'Console' (containing 'Output Console'), and 'Map' (containing 'Map'). The 'Scripts' panel lists various scripts like 'Pixel Lon Lat', 'Polynomial', 'Zero Crossing', 'Image Collection', 'Clipped Composite', 'Expression Map', 'Filtered Composite', 'Linear Fit', 'Modis Cloud Masking', 'Simple Cloud Score', 'Landsat Simple Composite', 'Feature Collection', and 'Charts'. The 'Code Editor' panel shows a JavaScript script for computing the trend of nighttime lights from DMSP. The 'Inspector' panel shows the output of the script, including a point, pixels, and objects. The 'Map' panel shows a satellite view of a city with a heatmap overlay.

**API Docs**

**Your Data**

**Search**

**Your Code**

**Data Inspector**

**Output Console**

**Batch Tasks**

**Map**

**Your Scripts & Example Scripts**

**Drawing Tools**

```
1 // Compute the trend of nighttime lights from DMSP.
2
3 // Add a band containing image date as years since 1991.
4 function createTimeBand(img) {
5   var year = ee.Date(img.get('system:time_start')).get('year').subtract(1991);
6   return ee.Image(year).byte().addBands(img);
7 }
8
9 // Fit a linear trend to the nighttime lights collection.
10 var collection = ee.ImageCollection('NOAA/DMSP-OLS/NIGHTTIME_LIGHTS')
11   .select('stable_lights')
12   .map(createTimeBand);
13 var fit = collection.reduce(ee.Reducer.LinearFit());
14
15 // Display a single image
16 Map.setCenter(30, 45, 4);
17 Map.addLayer(fit);
```

Point (33.4, 47.99) at 10/10/10  
- Pixels  
- stable\_lights first asset: Image (1 band)  
stable\_lights: 36  
- stable\_lights trends: Image (2 bands) [x]  
scale: -0.2167524834867634  
offset: 26.011442242679564  
- Objects

[code.earthengine.google.com](https://code.earthengine.google.com)