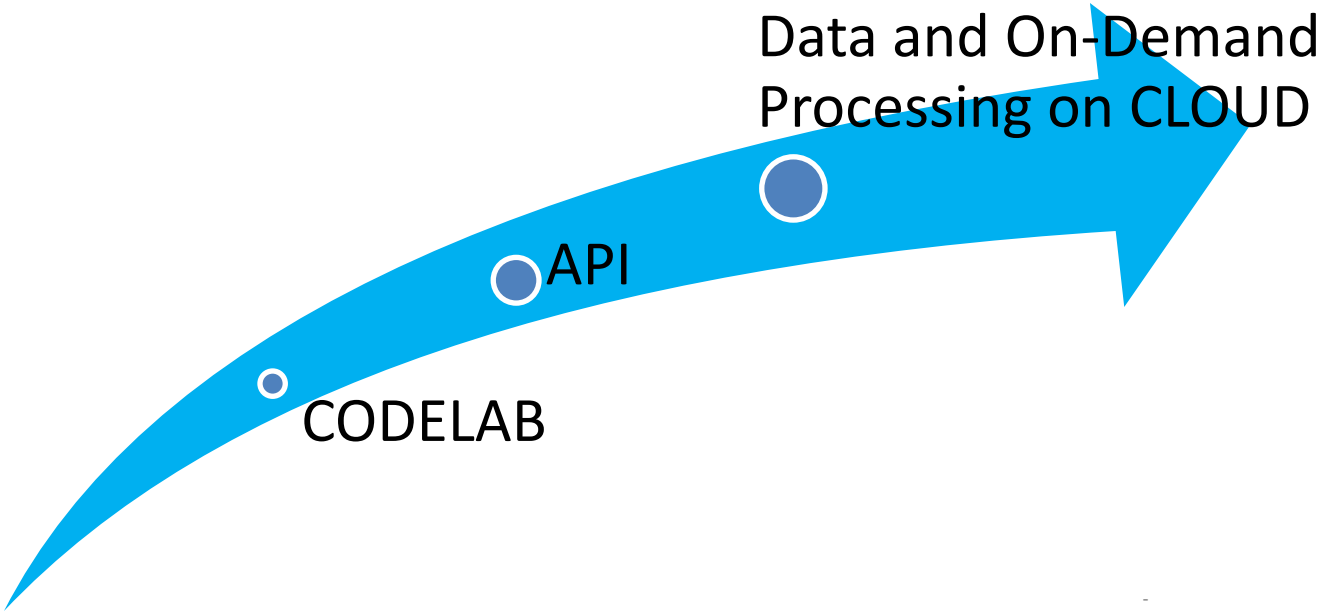
Bhoonidhi Upcoming Features



K.Radha Krishna
Bhoonidhi Team

Bhoonidhi: API

Bhoonidhi API to enable automatic machine to machine satellite data exchange.

- Spatio-temporal asset catalogue based API
- OPEN API specification
- SDKs for Python, java (Bhoonidhi-API-Lib)

Bhoonidhi - API Endpoints

default

GET	/	Landing Page
GET	/conformance	Conformance Classes
GET	/collections/{collection_id}/items/{item_id}	Get Item
GET	/search	Search
POST	/search	Search
GET	/collections	Get Collections
GET	/collections/{collection_id}	Get Collection
GET	/collections/{collection_id}/items	Get Itemcollection

Transaction Extension

PUT	/collections/{collection_id}/items/{item_id}	Update Item
DELETE	/collections/{collection_id}/items/{item_id}	Delete Item
PUT	/collections	Update Collection
POST	/collections	Create Collection
DELETE	/collections/{collection_id}	Delete Collection
POST	/collections/{collection_id}/items	Create Item

IRS data on STAC explorer

RS2_LIS3_-_16-JUL-2021_48_64_x_SAN_53147_16-JUL-2021_SSRN_53142_1_000_432_SAN_SSRN_1_53147_SAN_SSRN_1_53147_1

Source | Share

in stac-fastapi | Go to Parent | Go to Collection | Browse | Search

Collection

01/02/2016, 0:00:00 UTC - 12/12/2022, 0:00:00 UTC

RS2_LIS3

Bhoonidhi RS2_LIS3 Collection

Metadata

General

YAW	2.823
ROLL	-0.173
PITCH	-0.049
ROWNO	64
PATHNO	48
SENSOR	LIS3
BAND NOS	432
BOUNDARY	- - 22.37 - 13.885 - - 23.662 - 13.622 - - 23.368

Bhoonidhi : Codelab

- Compute resource provided through web – Jupyter notebook interface
- Limited default Storage provided to every user

- Bhoonidhi library to access data using the API (sdk)
- Visualization directly inside the Jupyter notebook

Bhoonidhi Codelab - Manage Jupyter Notebook

Deriving a vegetation index from Bhoonidhi - Sentinel2 data ¶

Researchers often use a vegetation index called NDVI to measure the "greeenness" or density of vegetation across a landscape. In addition to monitoring vegetation health, NDVI (Normalized Difference Vegetation Index) can be used to track climate change, agricultural production, desertification, and land cover change. Developed by NASA scientist Compton Tucker in 1977, NDVI is derived from satellite imagery and compares reflected near-infrared light to reflected visible red light.

In general, healthy and/or dense vegetation reflects a lot of near-infrared light and not as much red visible light. Conversely, when vegetation is sparse or not-so-healthy, its near-infrared reflectance decreases and its red light reflectance increases.

In this guide, you'll perform a basic NDVI calculation on Sentinel imagery using just a few lines of Python. Here are the steps:

1. Download a Sentinel Image from Bhoonidhi
2. Extract data from the red and near-infrared bands
3. Normalize the data
4. Perform the NDVI calculation
5. Save the NDVI image
6. Visualize the images

In [1]:
```
from codelab.bhoonidhi import Bhoonidhi
from codelab.codelab import showonmap,swipeonmap,initviz
```

In [2]:
```
bh=Bhoonidhi('browse','browse',site='staging')
bh.Login()
```

Bhoonidhi Codelab - Visualisation

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

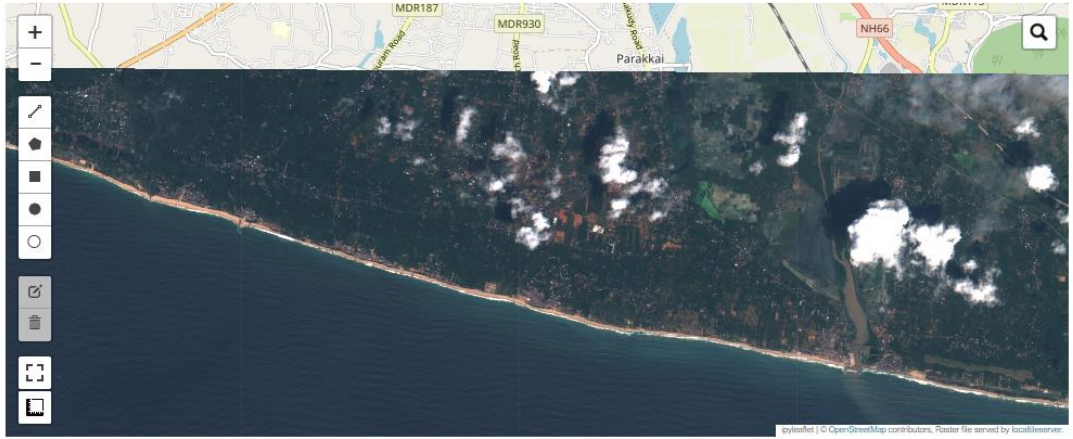
Run Code

```
import pprint

import datetime
import requests
import pprint
import json
import os
import ssl
jp2files = []
for dirpath, subdirs, files in os.walk('/home/jovyan'):
    for x in files:
        if x.endswith("_TCI.jp2"):
            jp2files.append(os.path.join(dirpath, x))

jp2files
```

Out[10]:
```
['/home/jovyan/S2A_MSIL1C_20221025T050901_N0400_R019_T43NGJ_20221025T070820.SAFE/GRANULE/L1C_T43NGJ_A038340_20221025T052128/IMG_DATA/T43NGJ_20221025T050901_TCI.jp2']
```

In [30]:
```
showonmap(jp2files[0])
```

Out[30]:



In [12]:
```
from osgeo import gdal,osr
import numpy as np
import os
```

On Demand Processing tool kit

1) Process published and verified algorithms with satellite imagery
2) Visualise , analyse and download outputs
3) Prepare and publish dashboards with periodic processing
4) Configure alerts on insights via mail or telegram

On demand processing tool kit

These are few basic tools in the kits along with other algorithms readily available at ODP.

1. On Demand Processing tool-kit to perform data processing tasks with the provision to download or visualize the output
2. Band wise access
3. AOI based sub-setting, mosaic, existing algorithms repository
4. AOI Based Segmentation
5. Raster mosaic
6. Raster calculation, querying Band arithmetic
7. Re-projection options

THANK YOU…