การสำรวจระยะไกลแบบออฟติคัล
และเข้าถึงได้แบบเปิด
Open Optical Remote Sensing

รองศาสตราจารย์ ดร ไพศาล สันติธรรมนนท์

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย

Email: Phisan.chula@gmail.com

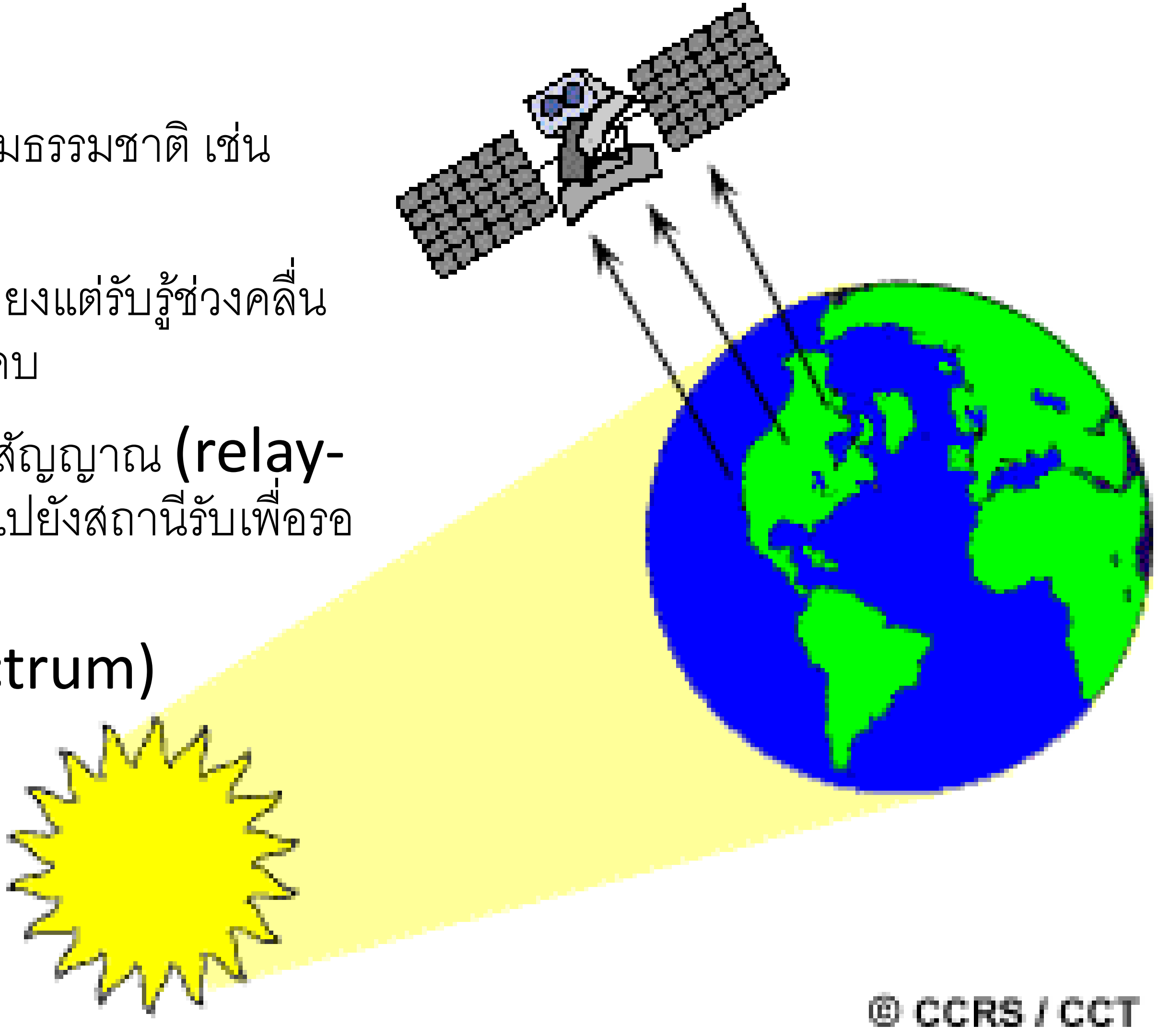
Since 2016 Open Remote Sensing

การสำรวจระยะไกล (Remote Sensing)

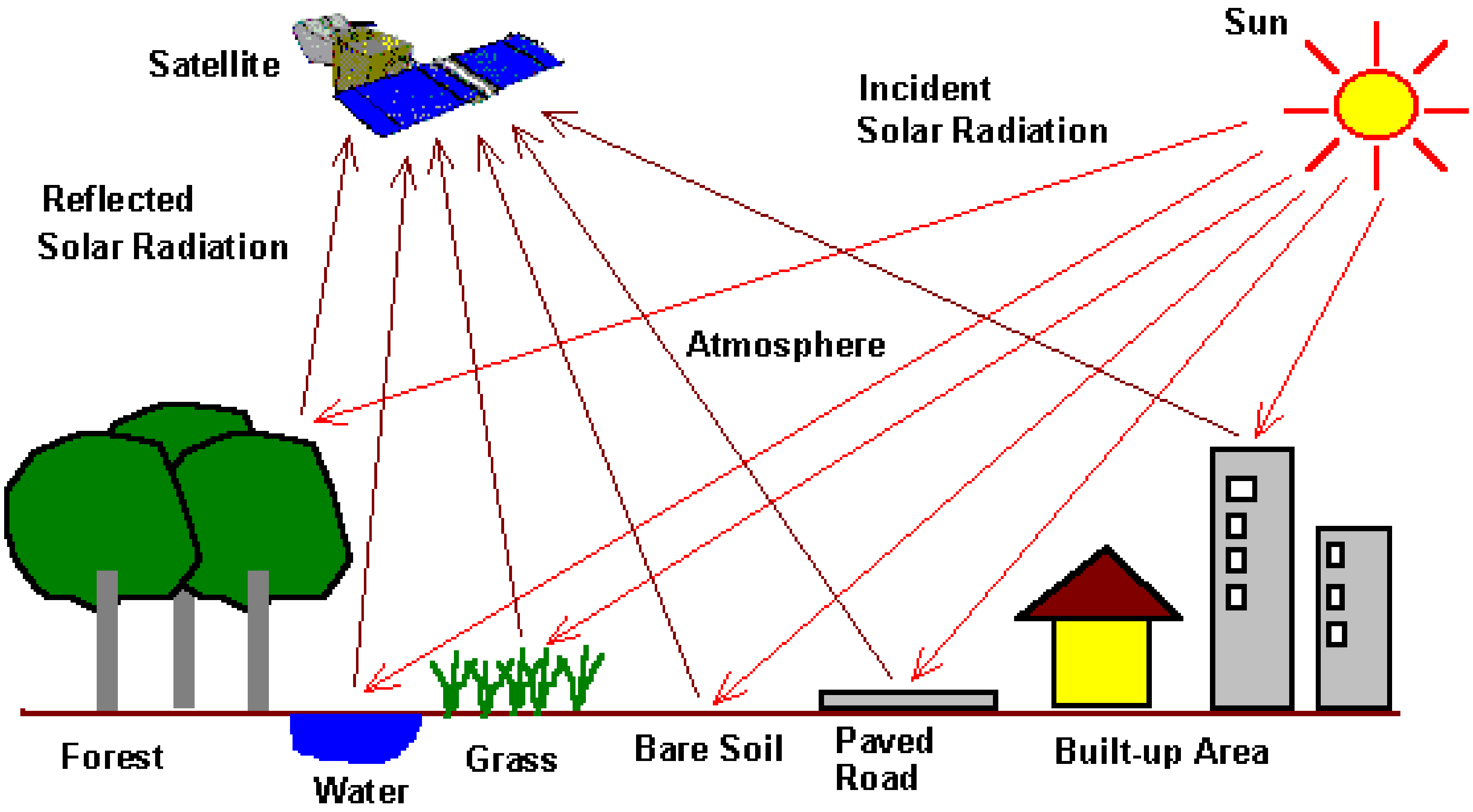
- การรับรู้จากระยะไกล (อังกฤษ: remote sensing) เป็นการได้มาซึ่งข้อมูลเกี่ยวกับวัตถุหรือปรากฏการณ์โดยปราศจากการสัมผัสทางกายภาพกับวัตถุหรือปรากฏการณ์นั้นๆ โดยตรง การรับรู้จากระยะไกลถูกนำมาใช้ในหลายสาขาโดยเฉพาะในสาขาภูมิศาสตร์ การสำรวจรังวัดที่ดินและวิทยาศาสตร์โลก เช่น อุทกวิทยา นิเวศวิทยา สมุทรศาสตร์ วิทยาธารน้ำแข็ง และธรณีวิทยา รวมถึงด้านการทหาร การสืบราชการลับ การค้า เศรษฐกิจ การวางแผน และการใช้งานด้านมนุษยธรรม
- ในปัจจุบันการรับรู้จากระยะไกลโดยทั่วไปหมายถึงการใช้เทคโนโลยีการรับรู้ทางอากาศในการตรวจสอบและจำแนกประเภทวัตถุบนโลกทั้งบนพื้นผิว มหาสมุทร และชั้นบรรยากาศของโลกด้วยหลักของการกระจายของคลื่น เช่น คลื่นแม่เหล็กไฟฟ้า ซึ่งอาจแบ่งประเภทออกเป็นเป็นการรับรู้ระยะไกลแบบใช้พลังงานในตัวเอง (active remote sensing) ที่มักจะเป็นระบบ radar เมื่ออากาศยานหรือดาวเทียมมีการปล่อยสัญญาณด้วยตัวเอง หรือการรับรู้ระยะไกลแบบไม่ใช้พลังงานในตัวเอง (passive remote sensing) ที่มักจะเป็นระบบ optical โดยได้รับข้อมูลจากการสะท้อนของพลังงานอื่น ๆ เช่น แสงอาทิตย์
- จากวิกิพีเดีย สารานุกรมเสรี

Optical / Passive Remote Sensing

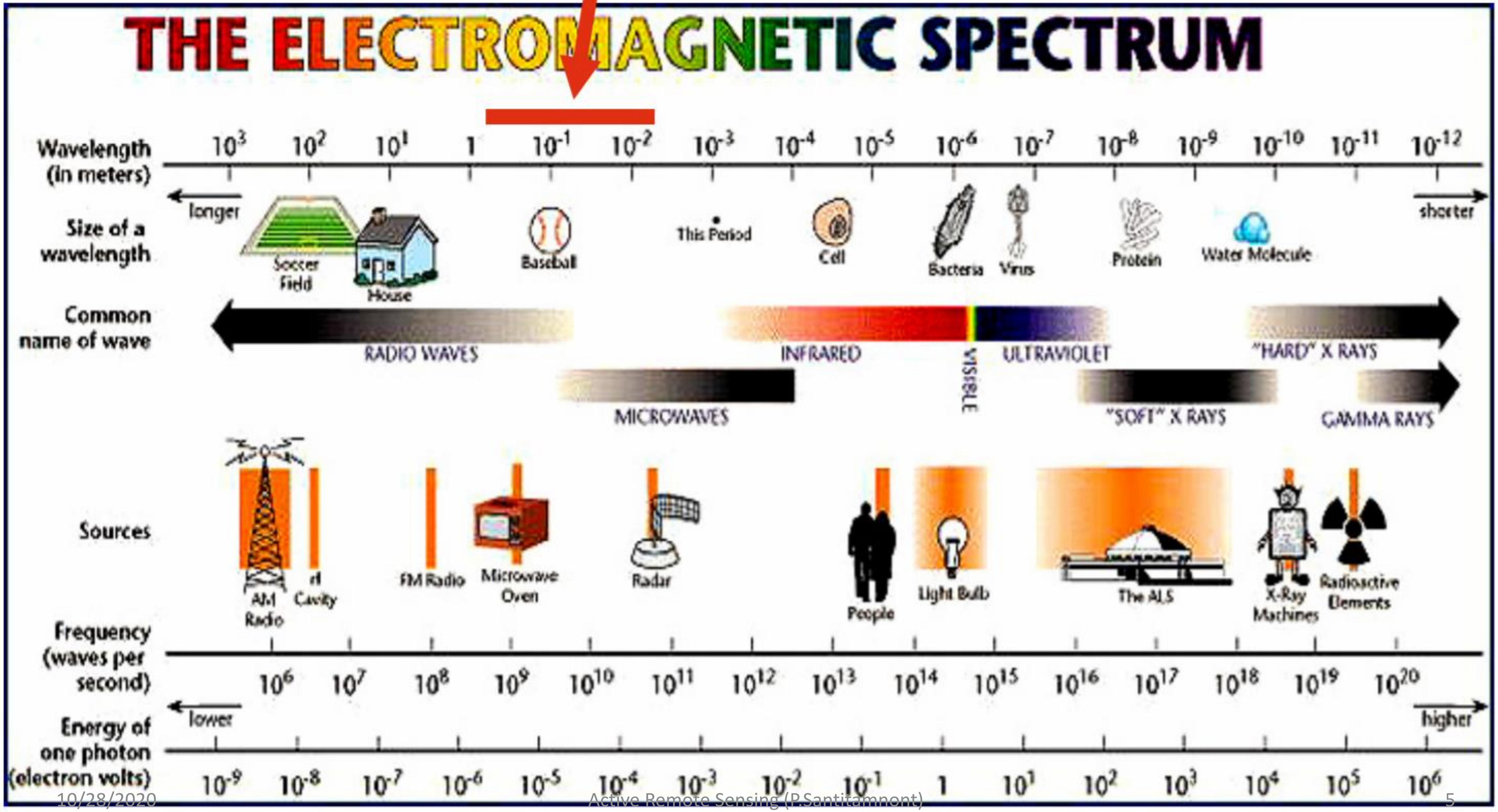
- การรับรู้ที่อาศัยการสะท้อนหรือการปรากฏของพลังงานตามธรรมชาติ เช่น แสงอาทิตย์ คลื่นความร้อน
- เซนเซอร์ติดตั้งบนดาวเทียมไม่ได้ส่งพลังงานใดๆออกมา เพียงแต่รับรู้ช่วงคลื่นในคุณลักษณะต่างๆ เช่น ช่วงคลื่นที่ความถี่ต่างๆ ในช่วงแคบ
- เมื่อดาวเทียมโคจรผ่านสถานีรับ หรืออาจมีดาวเทียวส่งต่อสัญญาณ (relay-satellite) ข้อมูลหรือภาพ จะทำการถ่ายทอดข้อมูลลงไปยังสถานีรับเพื่อรอให้บริการแก่ผู้ใช้ต่อไป
- ข้อมูลที่ได้มักจะเป็น ข้อมูลมัลติสเปคตรัม (multispectrum)
- ตัวอย่างระบบดาวเทียม
    - ระบบดาวเทียมแลนด์แซท L7 และ L8 (L9 etc …)
    - ระบบดาวเทียมเซนทินัล 2A และ 2B



Optical Remote Sensing Scenario

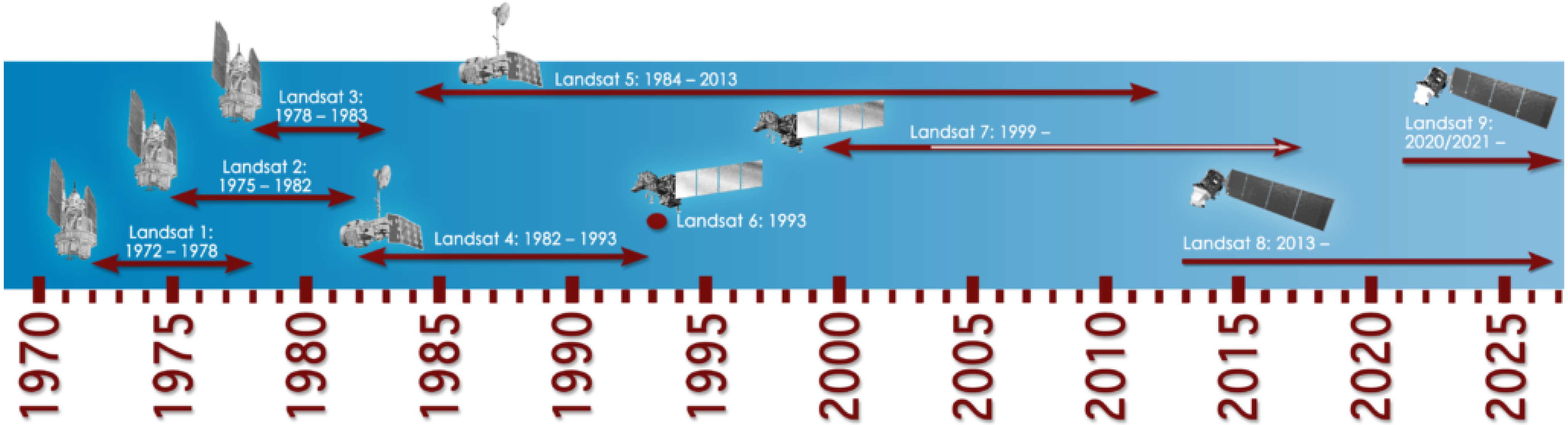


Radar remote sensing



NASA Landsat Program

- Landsat 8 launched on Feb. 11, 2013
- Landsat 9 [will] propel the program past 50 years of collecting global land cover data



Landsat 8

- Launch date 11 February 2013, 18:02 UTC
- Sensor : Operational Land Image (OLI)

  Thermal InfraRed Sensor (TIRS)

Spectral Band	Wavelength	Resolution	Solar Irradiance
Band 1 - Coastal / Aerosol	0.433 – 0.453 μm	30 m	2031 W/(m^{2}μm)
Band 2 - Blue	0.450 – 0.515 μm	30 m	1925 W/(m^{2}μm)
Band 3 - Green	0.525 – 0.600 μm	30 m	1826 W/(m^{2}μm)
Band 4 - Red	0.630 – 0.680 μm	30 m	1574 W/(m^{2}μm)
Band 5 - Near Infrared	0.845 – 0.885 μm	30 m	955 W/(m^{2}μm)
Band 6 - Short Wavelength Infrared	1.560 – 1.660 μm	30 m	242 W/(m^{2}μm)
Band 7 - Short Wavelength Infrared	2.100 – 2.300 μm	30 m	82.5
Band 8 - Panchromatic	0.500 – 0.680 μm	15 m	1739
Band 9 - Cirrus	1.360 – 1.390 μm	30 m	361

OLI Spectral Bands [11][*citation needed*]

Spectral Band	Wavelength	Resolution
Band 10 - Long Wavelength Infrared	10.30 – 11.30 μm	100 m
Band 11 - Long Wavelength Infrared	11.50 – 12.50 μm	100 m

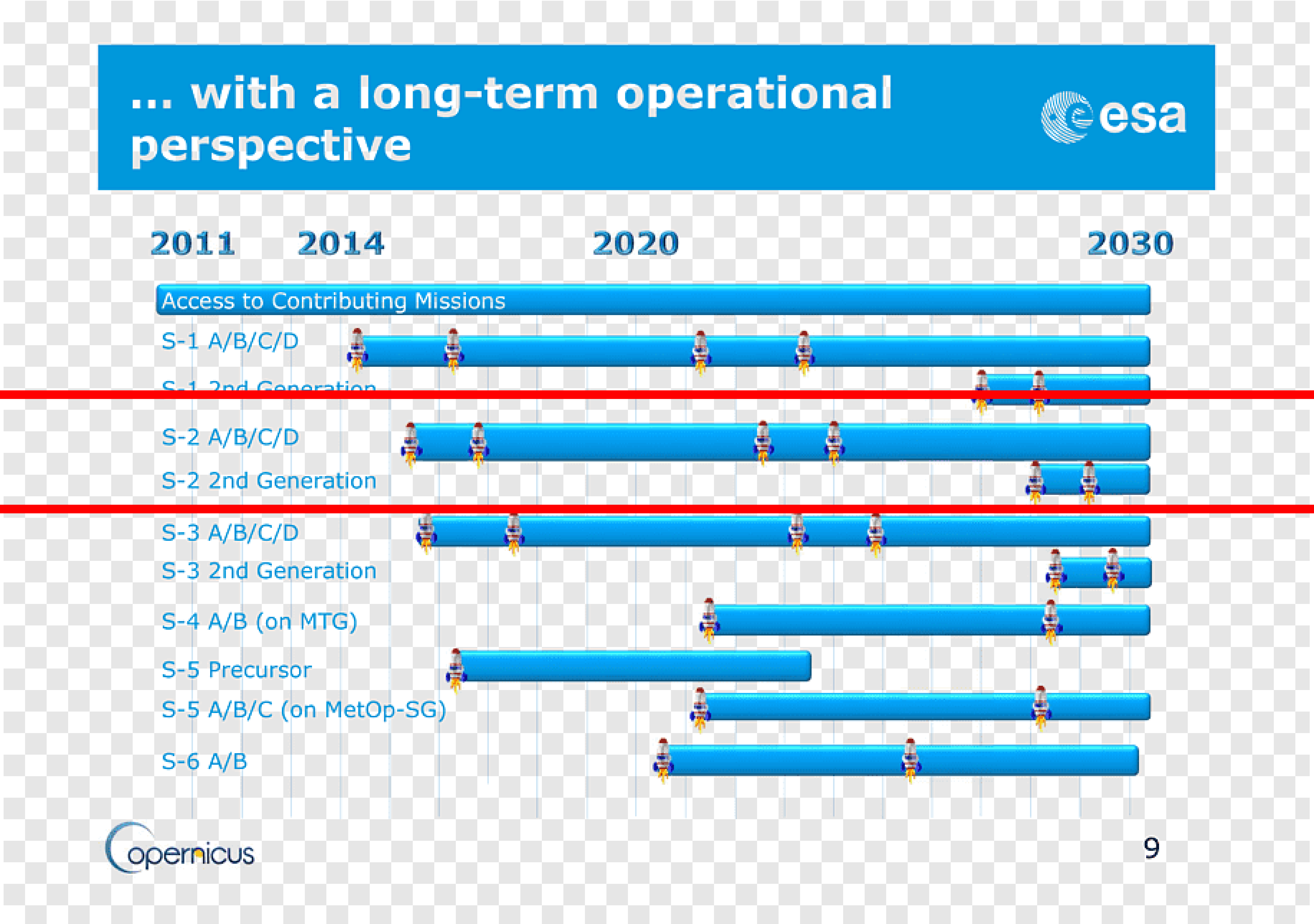
TIRS Spectral Bands [11]

Landsat8 OLI Usages

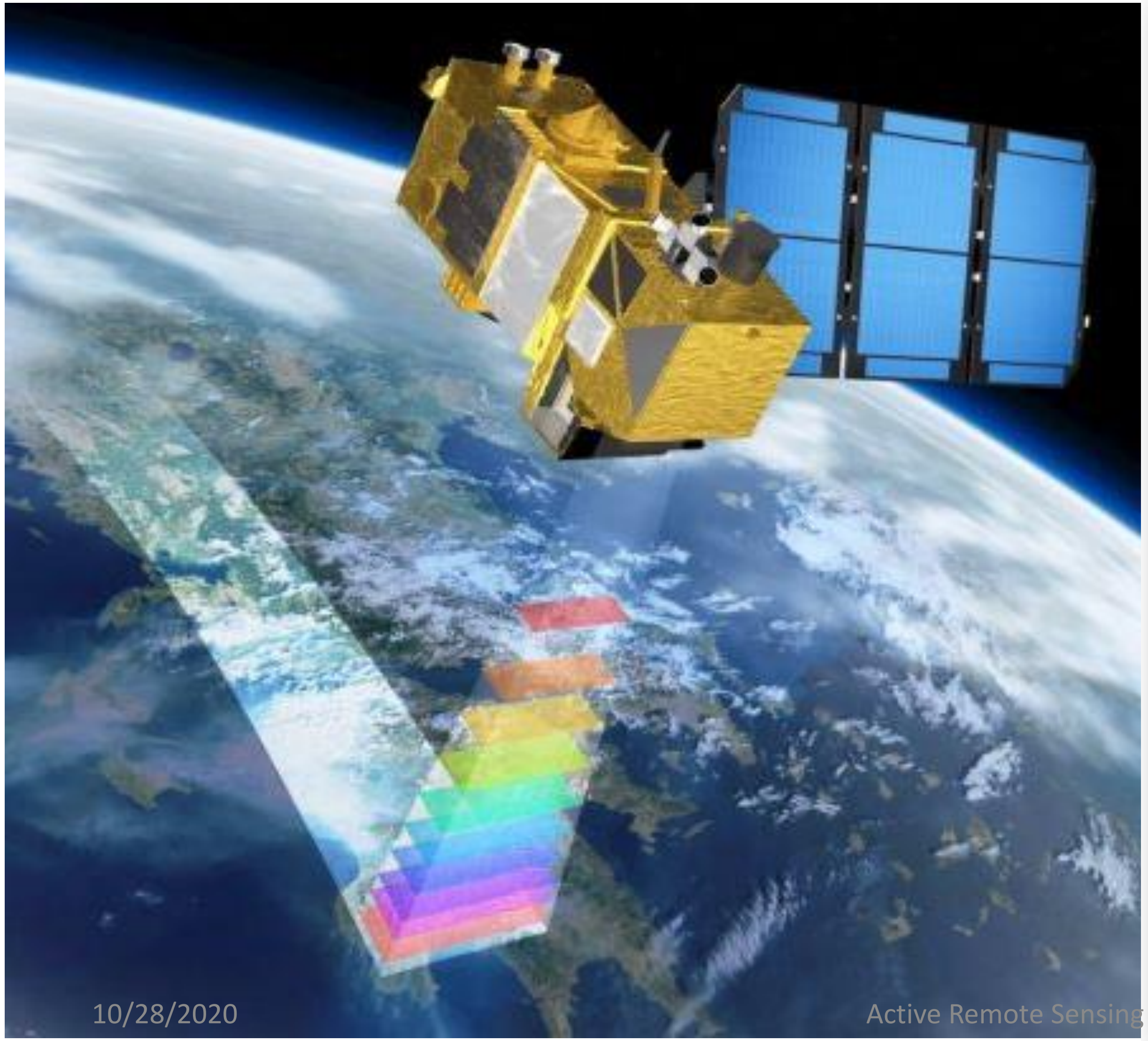
Band	Wavelength (μm)	Useful for mapping
Band 1: coastal aerosol	.43–.45	Coastal and aerosol studies
Band 2: blue	.45–.51	Bathymetric mapping, distinguishing soil from vegetation and deciduous from coniferous vegetation
Band 3: green	.53–.59	Emphasizes peak vegetation, which is useful for assessing plant vigour
Band 4: red	.64–.67	Discriminates vegetation slopes
Band 5: near-infrared (NIR)	.85–.88	Emphasizes biomass content and shorelines
Band6: short-wave infrared (SWIR) 1	1.57–1.65	Discriminates moisture content of soil and vegetation, penetrates thin clouds
Band 7: short-wave infrared (SWIR) 2	2.11–2.29	Improved moisture content of soil and vegetation and thin cloud penetration
Band 8: panchromatic	.50–.68	15 m resolution, sharper image definition
Band 9: cirrus	1.36–1.38	Improved detection of cirrus cloud contamination
Band 10: TIRS 1	10.60–11.19	100 m resolution, thermal mapping and estimated soil moisture
Band 11: TIRS 2	11.5–12.51	100 m resolution improved thermal mapping and estimated soil moisture

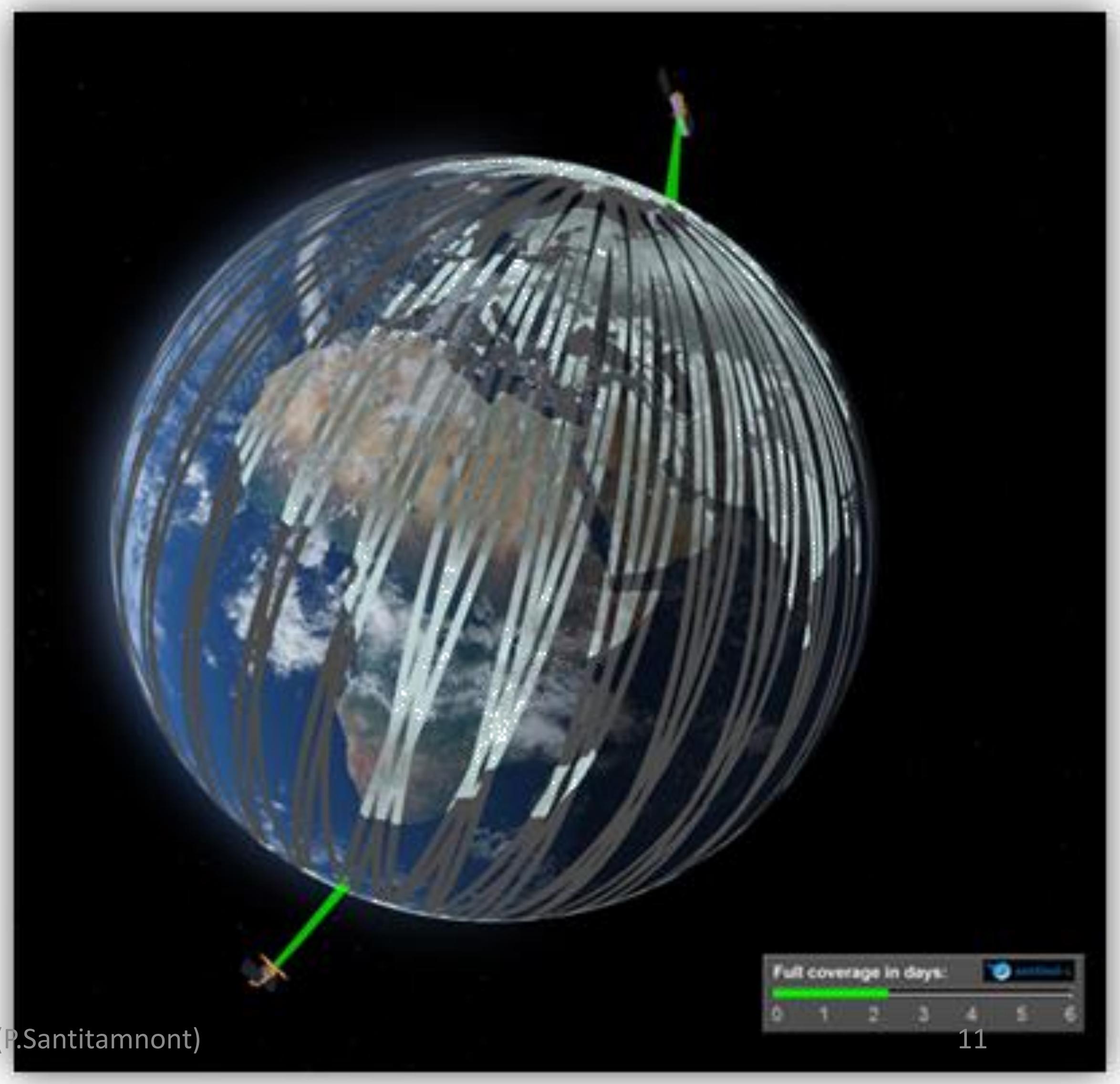
Sentinel 2A / 2B

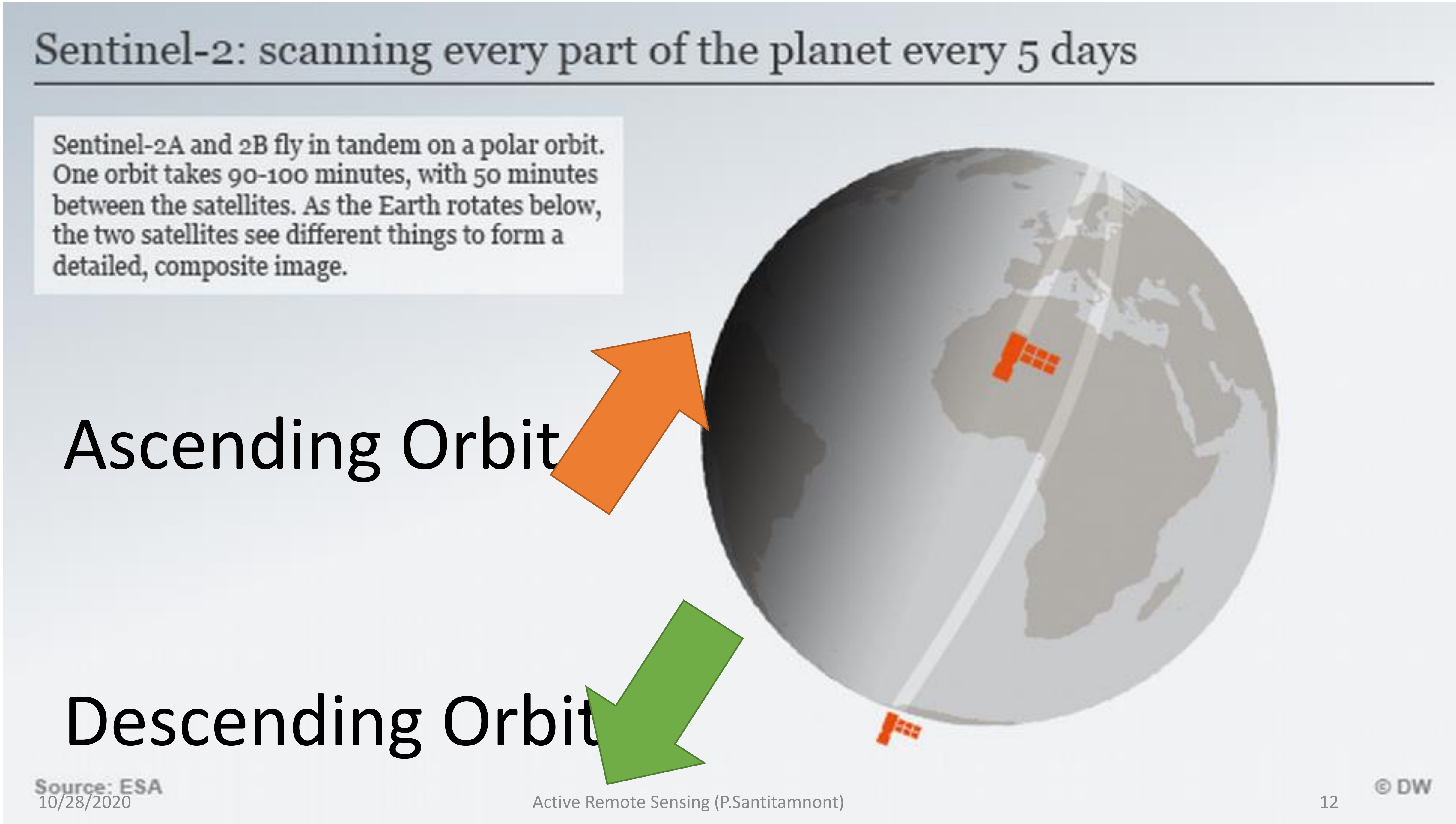
- Launch: 23 June 2015(2A)   7Mar2017 (2B)
- Multi-spectral data with 13 bands in the visible, near infrared, and short wave infrared part of the spectrum
- Revisiting every 5 days
- Spatial resolution of 10 m, 20 m and 60 m
- 290 km field of view
- **Free and open data policy**


... with a long-term operational perspective
esa
2011
2014
2020
2030
Access to Contributing Missions
S-1 A/B/C/D
S-1 2nd Generation
S-2 A/B/C/D
S-2 2nd Generation
S-3 A/B/C/D
S-3 2nd Generation
S-4 A/B (on MTG)
S-5 Precursor
S-5 A/B/C (on MetOp-SG)
S-6 A/B
copernicus
9

ESA Sentinel-2A / 2B







Ascending Orbit

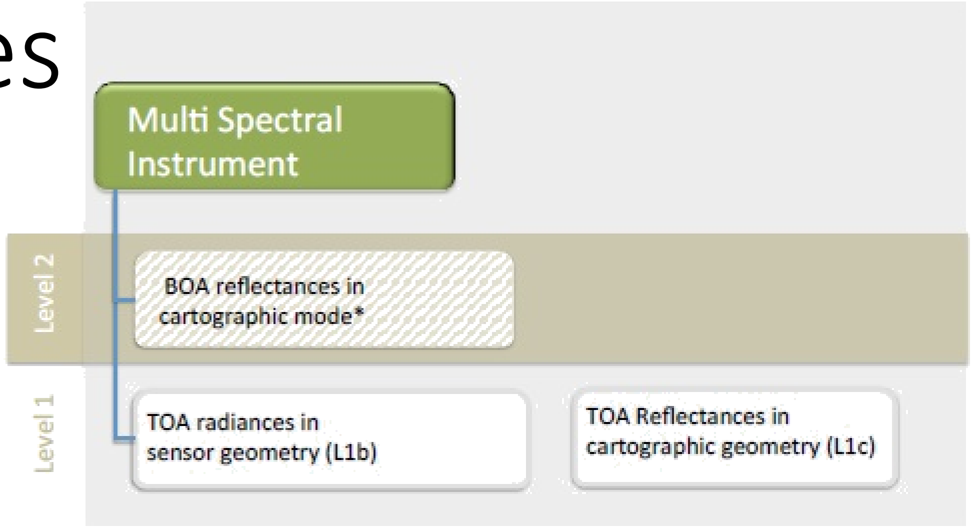
Descending Orbit

Spectral bands for the Sentinel-2 sensors[9]

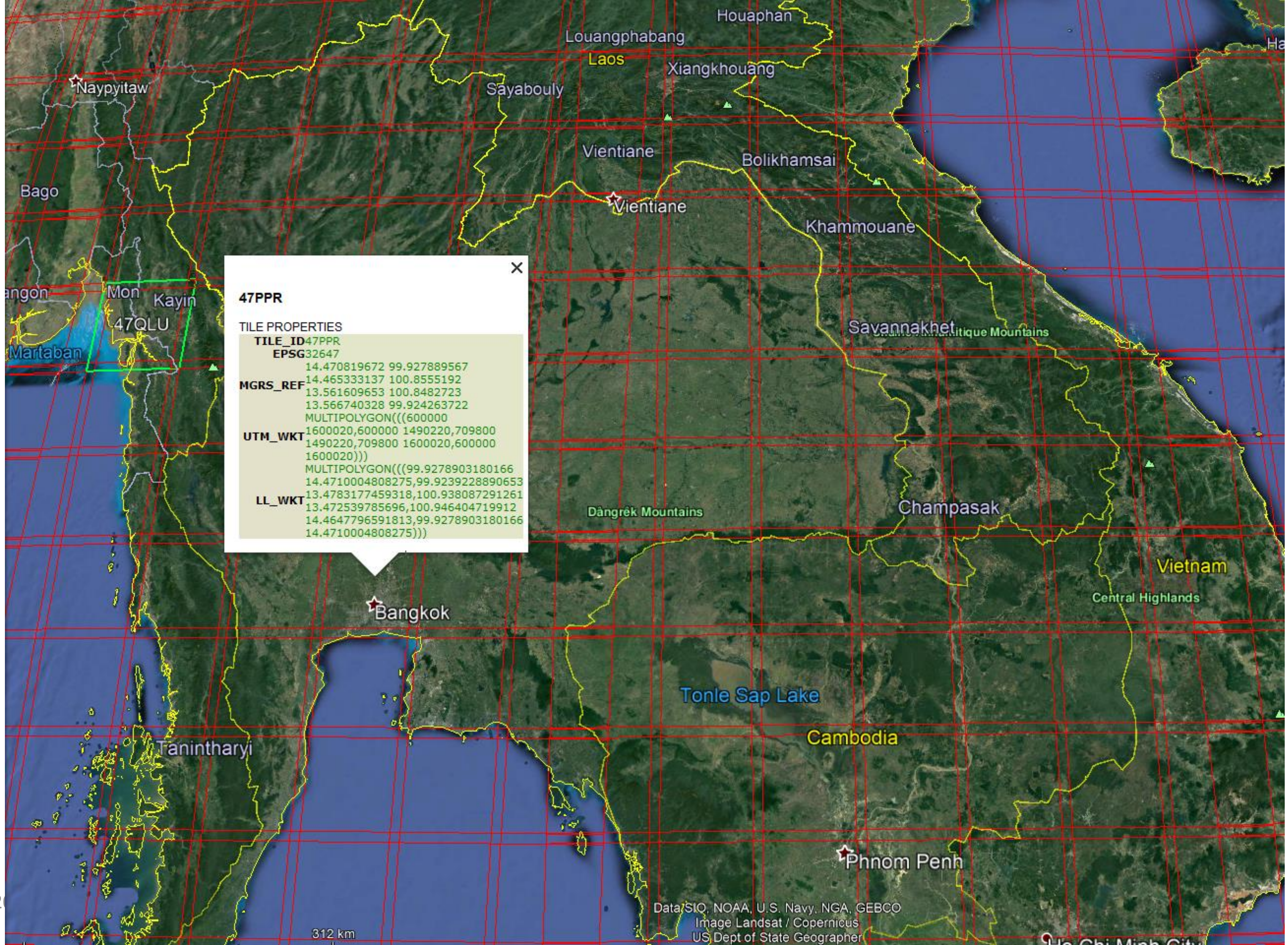
Sentinel-2 bands	Sentinel-2A		Sentinel-2B		
	Central wavelength (nm)	Bandwidth (nm)	Central wavelength (nm)	Bandwidth (nm)	Spatial resolution (m)
Band 1 – Coastal aerosol	442.7	21	442.2	21	60
Band 2 – Blue	492.4	66	492.1	66	10
Band 3 – Green	559.8	36	559.0	36	10
Band 4 – Red	664.6	31	664.9	31	10
Band 5 – Vegetation red edge	704.1	15	703.8	16	20
Band 6 – Vegetation red edge	740.5	15	739.1	15	20
Band 7 – Vegetation red edge	782.8	20	779.7	20	20
Band 8 – NIR	832.8	106	832.9	106	10
Band 8A – Narrow NIR	864.7	21	864.0	22	20
Band 9 – Water vapour	945.1	20	943.2	21	60
Band 10 – SWIR – Cirrus	1373.5	31	1376.9	30	60
Band 11 – SWIR	1613.7	91	1610.4	94	20
Band 12 – SWIR	2202.4	175	2185.7	185	20

Sentinel 2A/2B product types

- Granules / Tiles : 100x100km2 ortho-images in UTM/WGS84



Name	High-level Description	Production & Distribution	Data Volume
Level-1C	Top-of-atmosphere reflectances in cartographic geometry	Systematic generation and on-line distribution	600 MB (each 100x100 km2)
Level-2A	Bottom-of-atmosphere reflectance in cartographic geometry	Systematic generation and on-line distribution and generation on user side (using Sentinel-2 Toolbox)	800 MB (each 100x100 km2)


47PPR
TILE PROPERTIES
TILE_ID 47PPR
EPSG 32647
MGRS_REF 14.470819672 99.927889567 14.465333137 100.8555192 13.561609653 100.8482723 13.566740328 99.924263722
UTM_WKT MULTIPOLYGON(((600000 1600020,600000 1490220,709800 1490220,709800 1600020,600000 1600020)))
LL_WKT MULTIPOLYGON(((99.9278903180166 14.4710004808275,99.9239228890653 13.4783177459318,100.938087291261 13.472539785696,100.946404719912 14.4647796591813,99.9278903180166 14.4710004808275)))
47QLU
Naypyitaw
Bago
Mon
Kayin
Martaban
Tanintharyi
Louangphabang
Laos
Houaphan
Xiangkhouang
Sayabouly
Vientiane
Bolikhamsai
Khammouane
Savannakhet
Champasak
Dàngrêk Mountains
Bangkok
Tonle Sap Lake
Cambodia
Phnom Penh
Vietnam
Central Highlands
Data SIO, NOAA, U.S. Navy, NGA, GEBCO
Image Landsat / Copernicus
US Dept of State Geographer
312 km

การประยุกต์ใช้ multispectral image (passive)

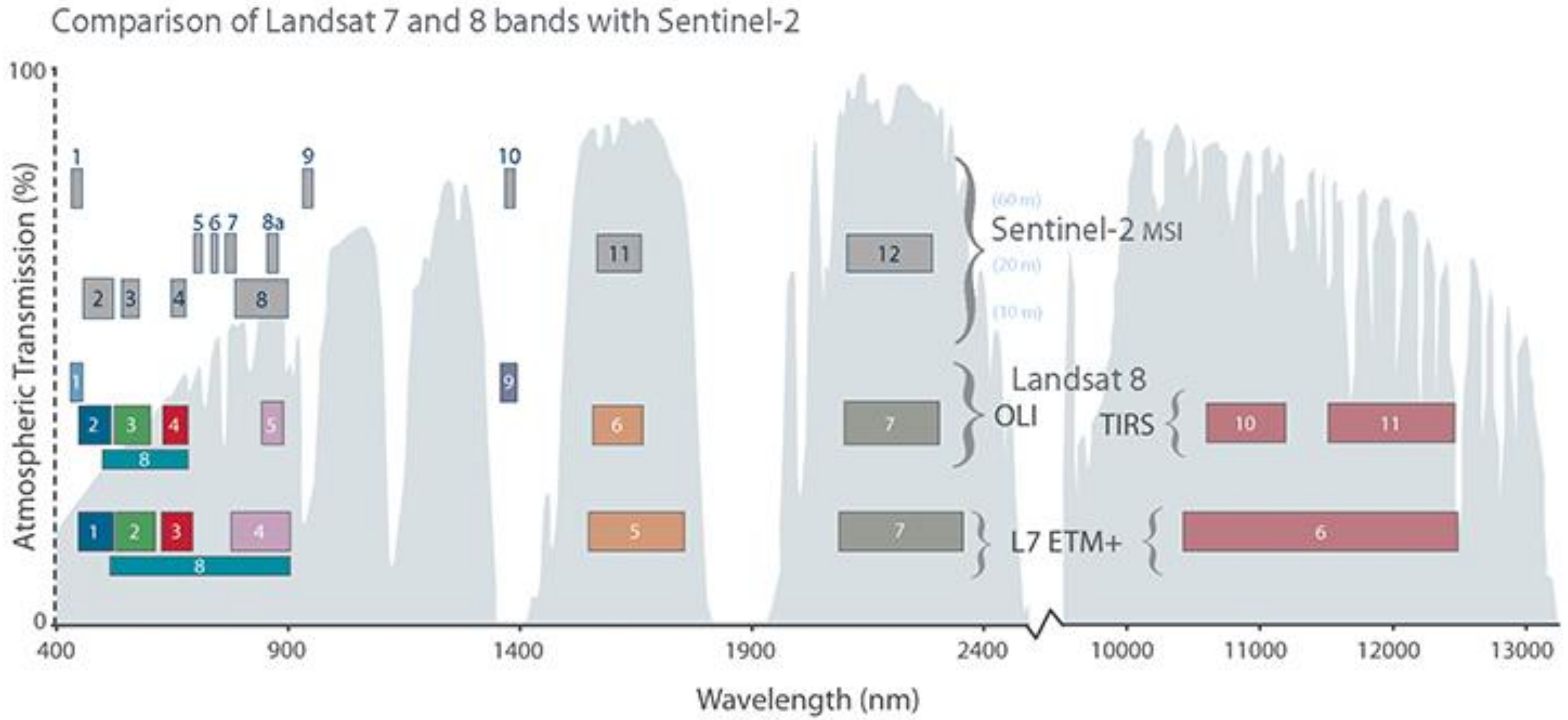
- แผนที่การใช้ประโยชน์ที่ดิน จาก color composites (true color / false color
- ดัชนี กายภาพ ชีวภาพ ของแผ่นดิน
  - Normalized Difference Vegetation Index (NDVI)
    - -1.0 to + 1.0
  - Normalized Difference RedEdge Index (NDRE)
- Normalized Difference Water Index (NDWI)
  - < 0.3 - Non-water >= 0.3 - Water.
- Normalised Difference Snow Index (NDSI)
- Normalized Burn Ratio (NBR)

custom-scripts

A repository of custom scripts that can be used with Sentinel-Hub services.

- Remote sensing indices
- Disaster management and prevention algorithms
- Agriculture and forestry algorithms
- Agriculture and forestry algorithms
- Urban planning algorithms

- https://custom-scripts.sentinel-hub.com/#sentinel-2


Comparison of Landsat 7 and 8 bands with Sentinel-2
Atmospheric Transmission (%)
100
0
1
9
10
5 6 7
8a
11
12
2
3
4
8
(60 m)
Sentinel-2 MSI
(20 m)
(10 m)
1
9
2
3
4
5
6
7
8
Landsat 8
OLI
TIRS
10
11
1
2
3
4
5
7
8
L7 ETM+
6
400
900
1400
1900
2400
10000
11000
12000
13000
Wavelength (nm)

Level-2A Algorithm Overview

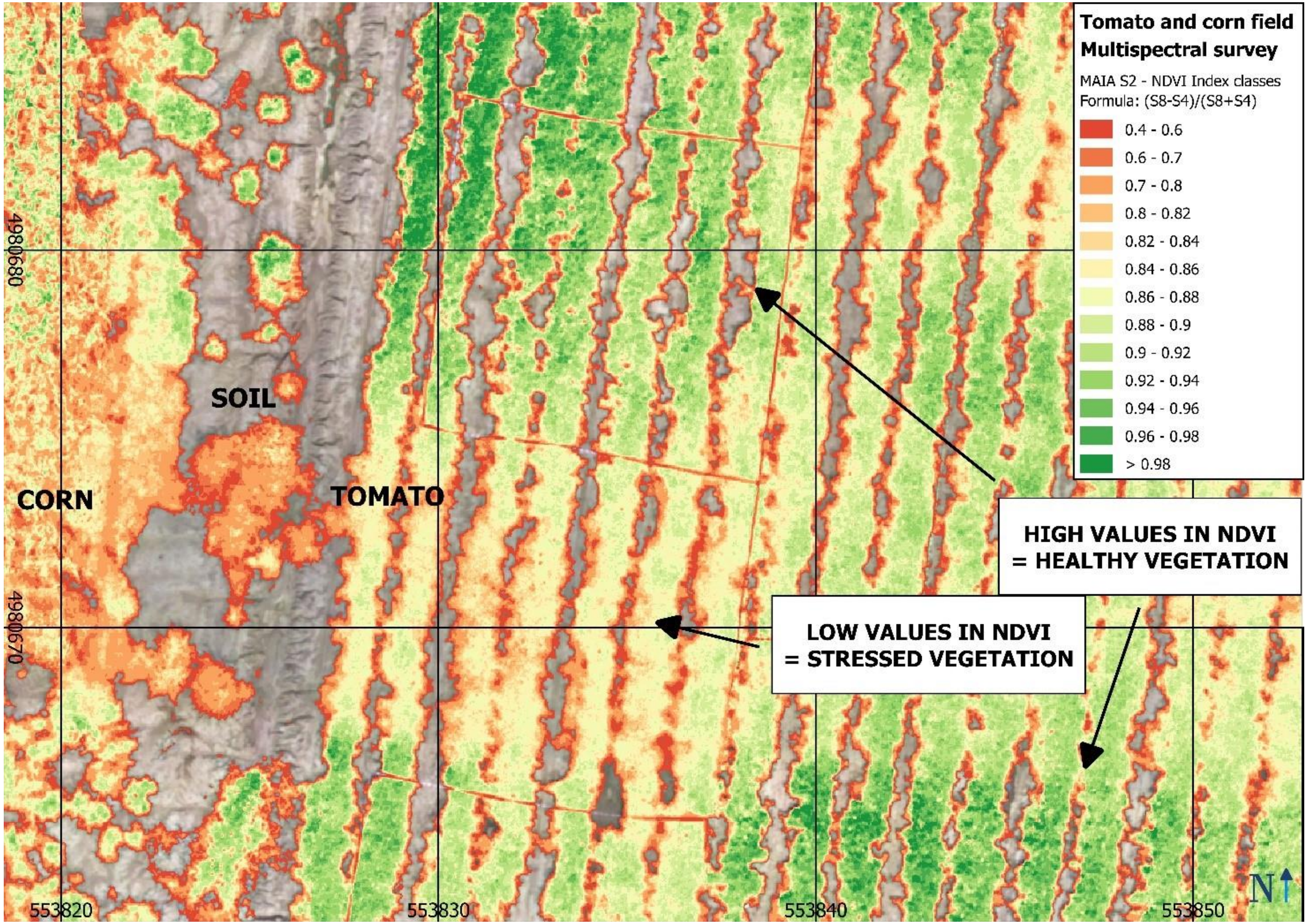
- Level-2A can also be generated by the User from the Level-1C product using the Sentinel-2 Toolbox or the standalone version of the Sen2Cor processor**.**
- Level 2A-processing is split into two parts:
  - Scene Classification (SC) aims at providing a pixel classification map (cloud, cloud shadows, vegetation, soils/deserts, water, snow, etc.)
  - Atmospheric Correction (S2AC) aims at transforming TOA reflectance into BOA reflectance.

- **From the middle of March 2018, the Level-2A became an operational product**

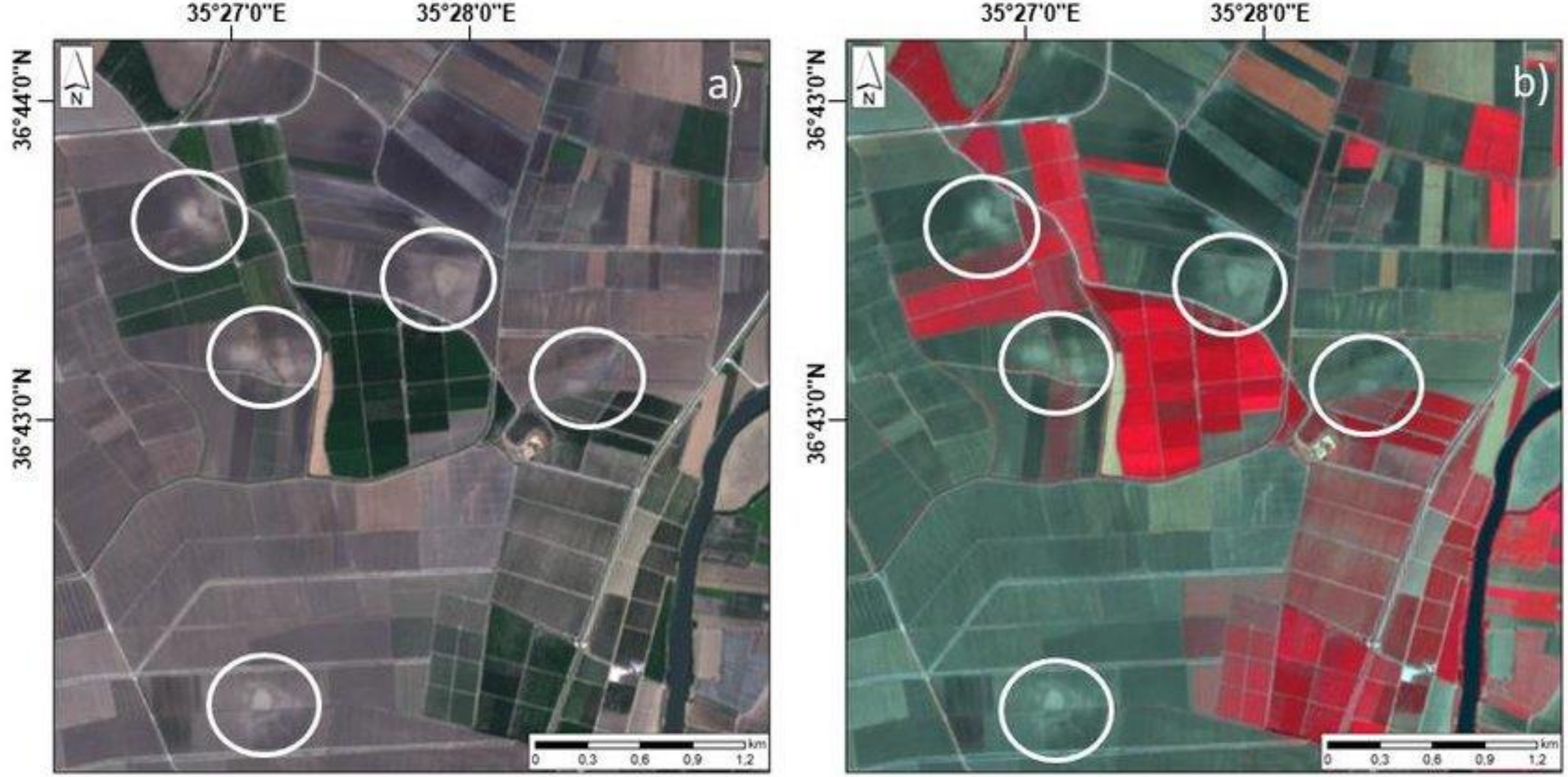
Classification Mask Generation

- SENTINEL-2 Level-1C product at 60 m resolution

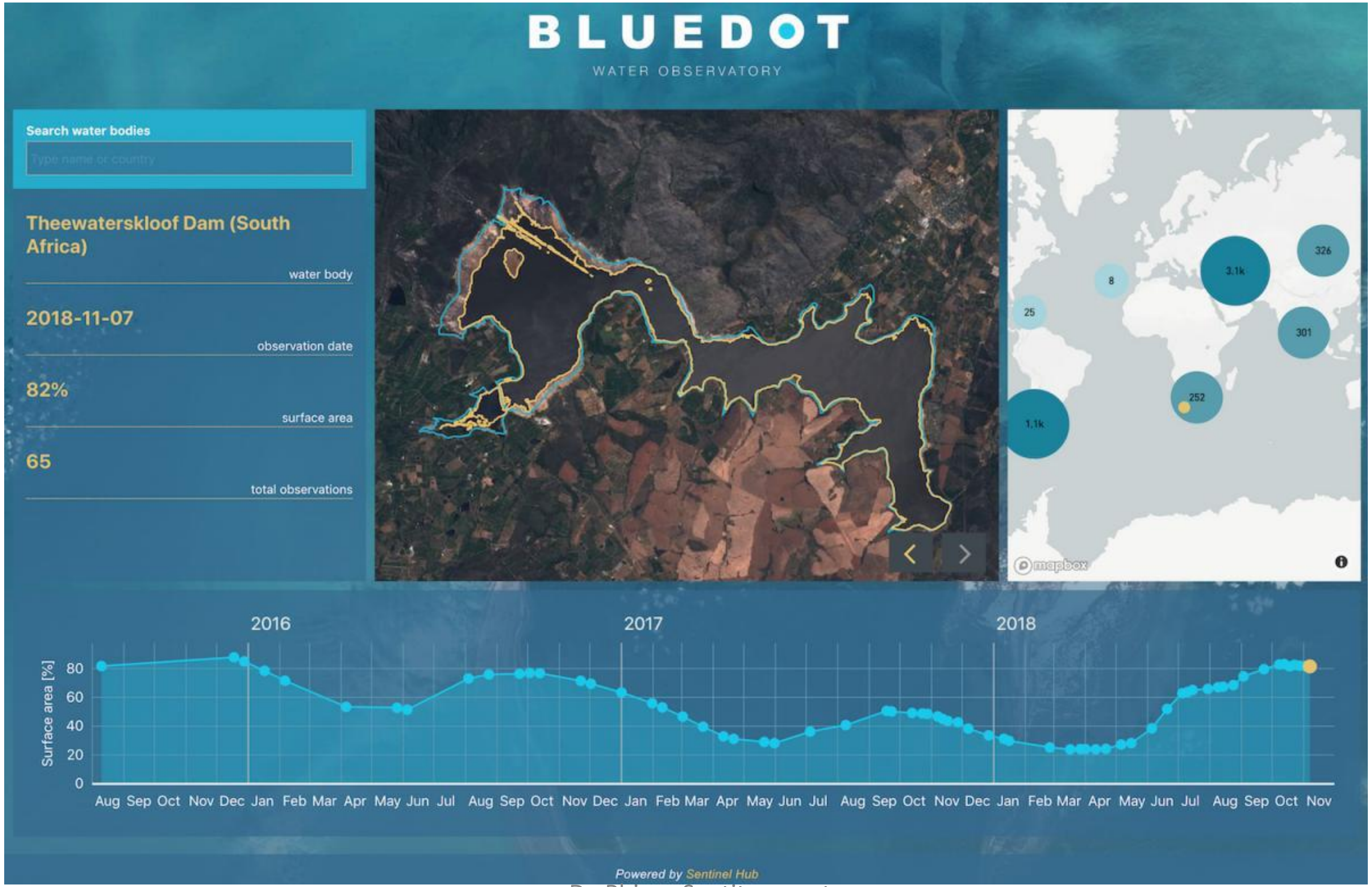
Label	Classification
0	NO_DATA
1	SATURATED_OR_DEFECTIVE
2	DARK_AREA_PIXELS
3	CLOUD_SHADOWS
4	VEGETATION
5	NOT_VEGETATED
6	WATER
7	UNCLASSIFIED
8	CLOUD_MEDIUM_PROBABILITY
9	CLOUD_HIGH_PROBABILITY
10	THIN_CIRRUS
11	SNOW


Tomato and corn field
Multispectral survey
MAIA S2 - NDVI Index classes
Formula: (S8-S4)/(S8+S4)
0.4 - 0.6
0.6 - 0.7
0.7 - 0.8
0.8 - 0.82
0.82 - 0.84
0.84 - 0.86
0.86 - 0.88
0.88 - 0.9
0.9 - 0.92
0.92 - 0.94
0.94 - 0.96
0.96 - 0.98
> 0.98
SOIL
CORN
TOMATO
HIGH VALUES IN NDVI
= HEALTHY VEGETATION
LOW VALUES IN NDVI
= STRESSED VEGETATION
4980680
4980670
553820
553830
553840
553850
N

Sentinel-2 satellite: (a) FCC 432 (bands 432 in RGB); and (b) FCC 843 (bands 843 in RGB respectively).

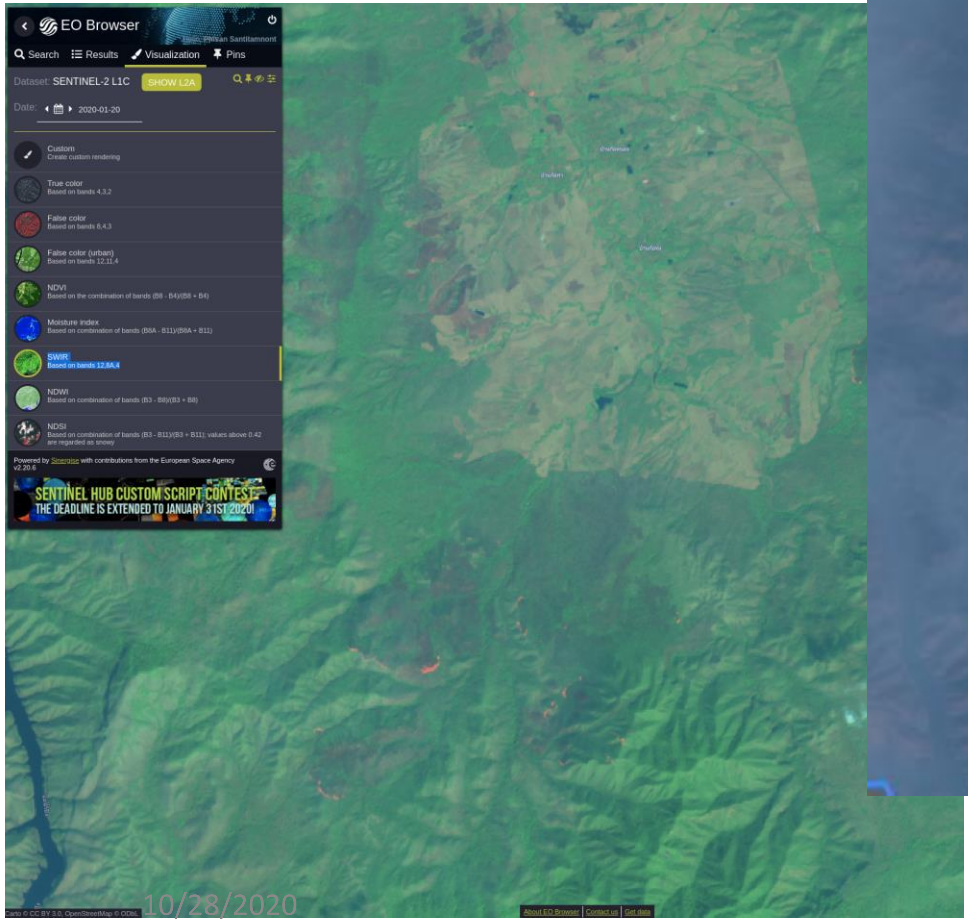


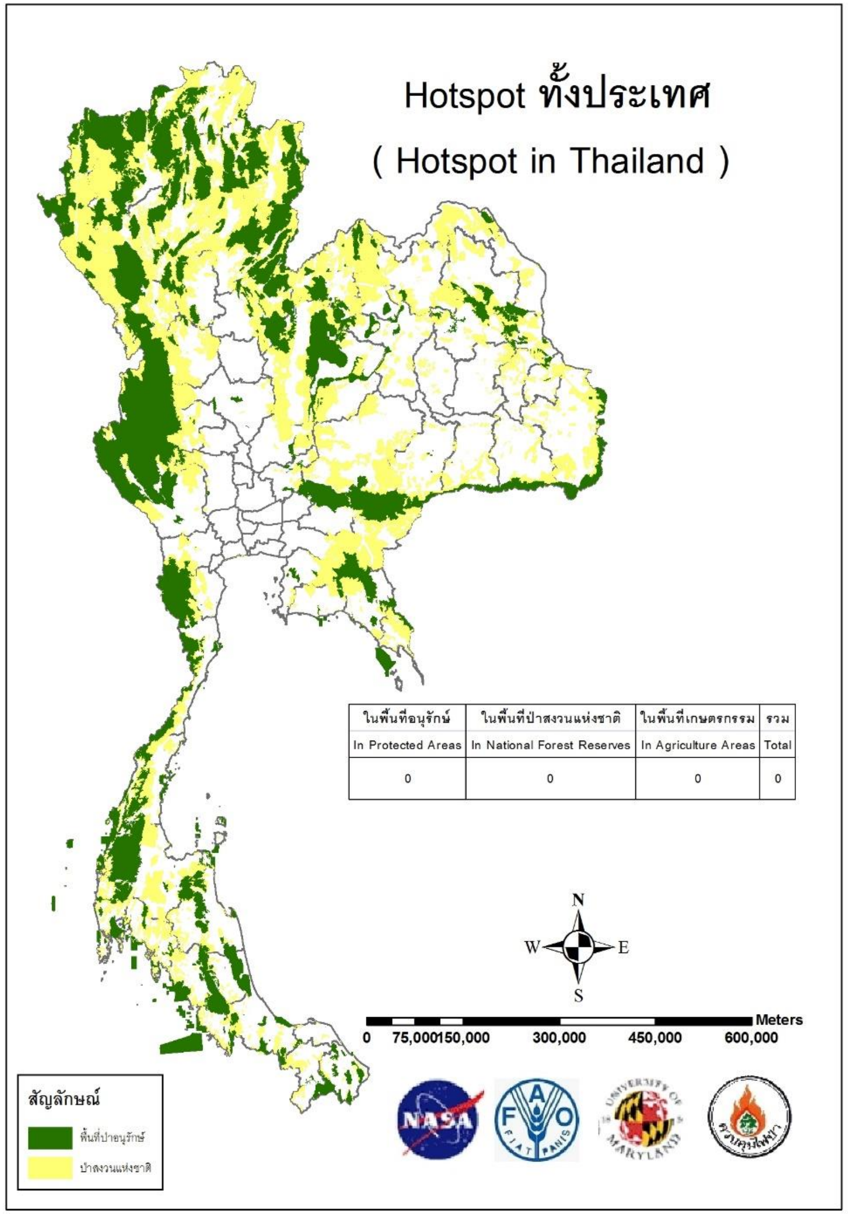
Series of Water Resource Monitoring (NDWI)



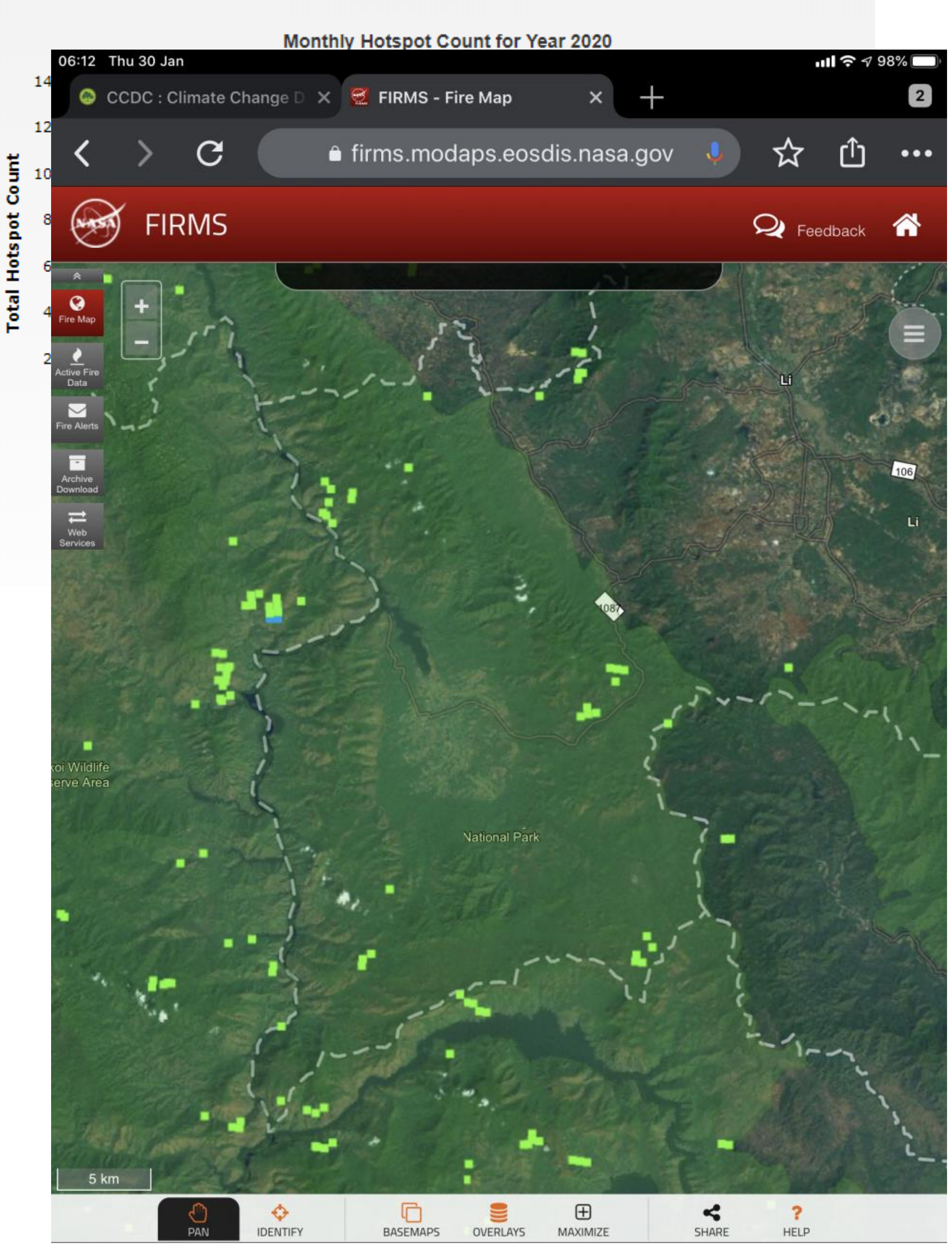
Wild Fire, Hot Spot and Fire Fighting (SWIR/NBR) …. Causing 70% of PM2.5

Wildfire Visualization via S2 (SWIR and others)





ในพื้นที่อนุรักษ์	ในพื้นที่ป่าสงวนแห่งชาติ	ในพื้นที่เกษตรกรรม	รวม
In Protected Areas	In National Forest Reserves	In Agriculture Areas	Total
0	0	0	0



Sentinel Hub

sentinelhub-js
Download and process satellite imagery in JavaScript using Sentinel Hub services.
sentinel · sentinel-2 · sentinel-hub · sentinel-1
TypeScript · MIT · 0 · 5 · 0 · 1 · Updated 10 hours ago

eo-learn
Earth observation processing framework for machine learning in Python
machine-learning · eo-data · python-package · eo-research
Python · MIT · 142 · 459 · 15 · 7 · Updated 2 days ago

eo-flow
Collection of TensorFlow 2.0 code for Earth Observation applications
Python · MIT · 4 · 12 · 0 · 0 · Updated 2 days ago

sentinel2-cloud-detector
Sentinel Hub Cloud Detector for Sentinel-2 images in Python
python-library · classification · sentinel-2 · sentinel-hub · cloud-detector
Python · CC-BY-SA-4.0 · 34 · 120 · 2 · 0 · Updated 6 days ago

sentinelhub-py
Download and process satellite imagery in Python scripts using Sentinel Hub

-snippets
on · 1 · 0 · 0 · 0 · Updated 13 days ago

-tool
y tool that prepares your data for use in Sentinel Hub
el-hub
MIT · 2 · 0 · 0 · 0 · Updated 15 days ago

r-observatory-backend
ring water levels of lakes and reservoirs using satellite imagery
er Notebook · MIT · 13 · 36 · 0 · 1 · Updated on Nov 3, 2019

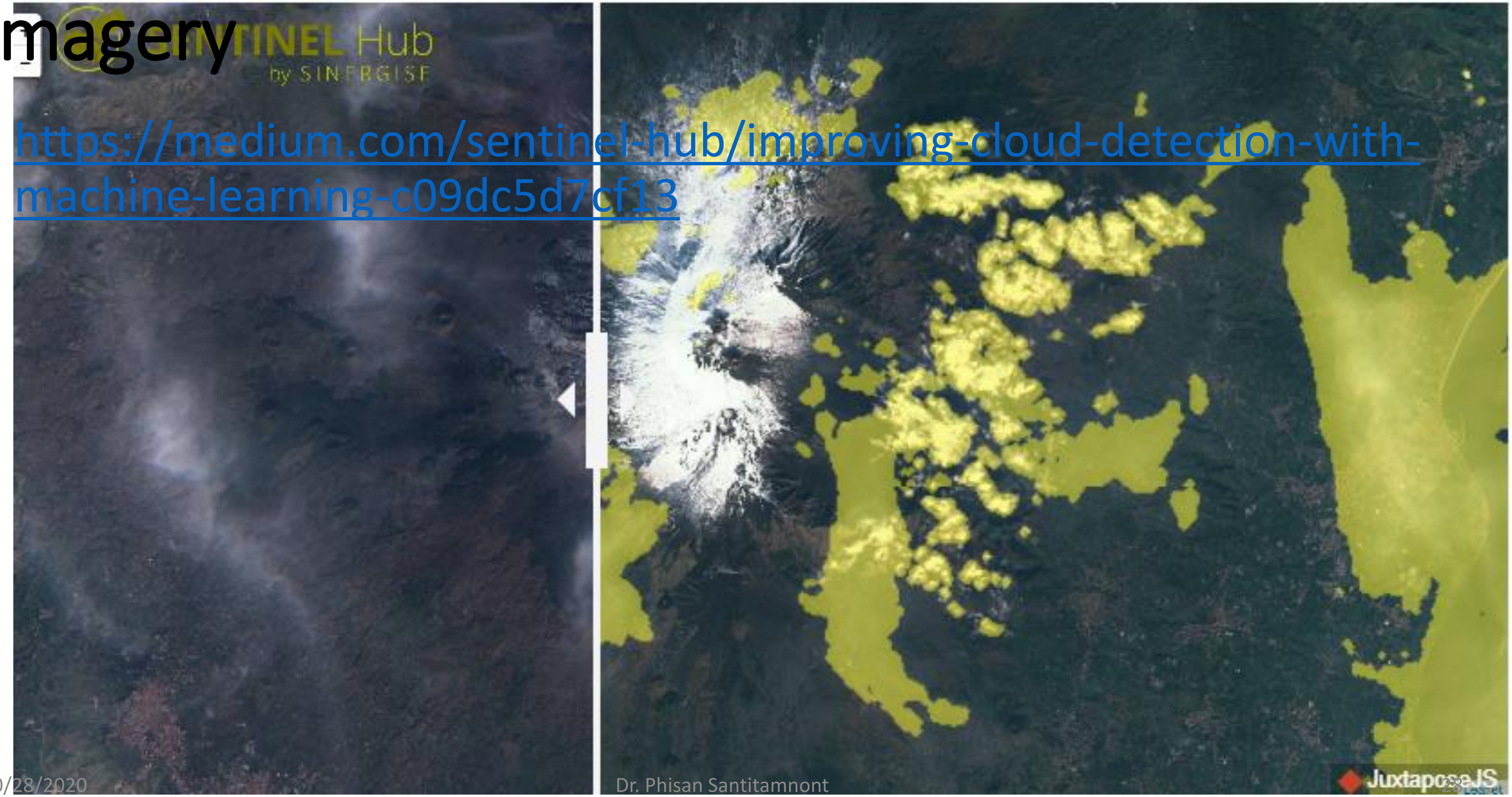
ification-app-frontend
cript · MIT · 0 · 1 · 0 · 1 · Updated on Nov 3, 2019

owser
rth Observation Browser is a search tool for Sentinel-1, -2, -3, Landsat 5, 7, is and Envisat satellite imagery
cript · MIT · 10 · 31 · 2 · 2 · Updated on Oct 29, 2019

inelPlayground

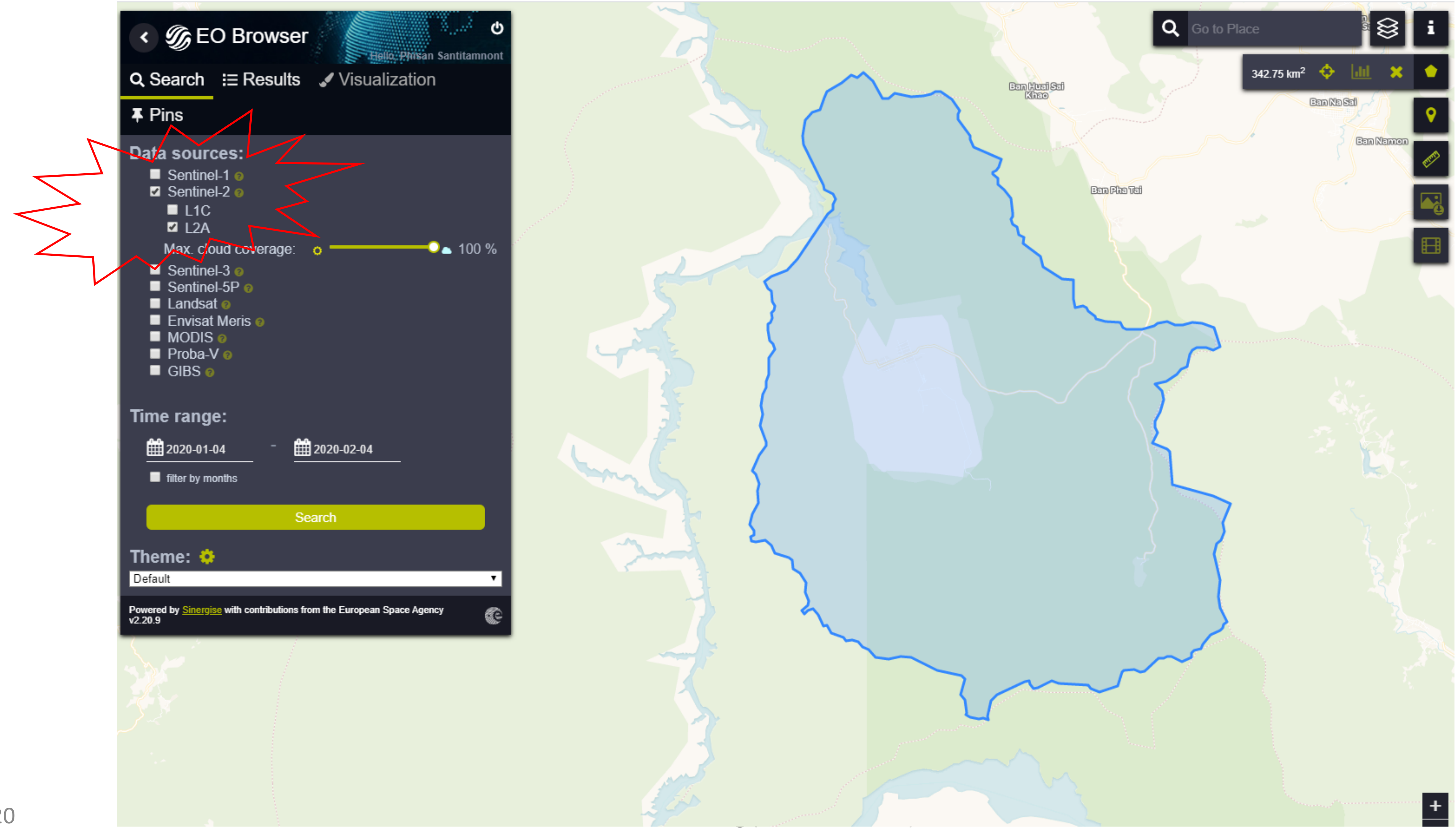
Sentinel Hub's cloud detector for Sentinel-2 imagery

- https://medium.com/sentinel-hub/improving-cloud-detection-with-machine-learning-c09dc5d7cf13



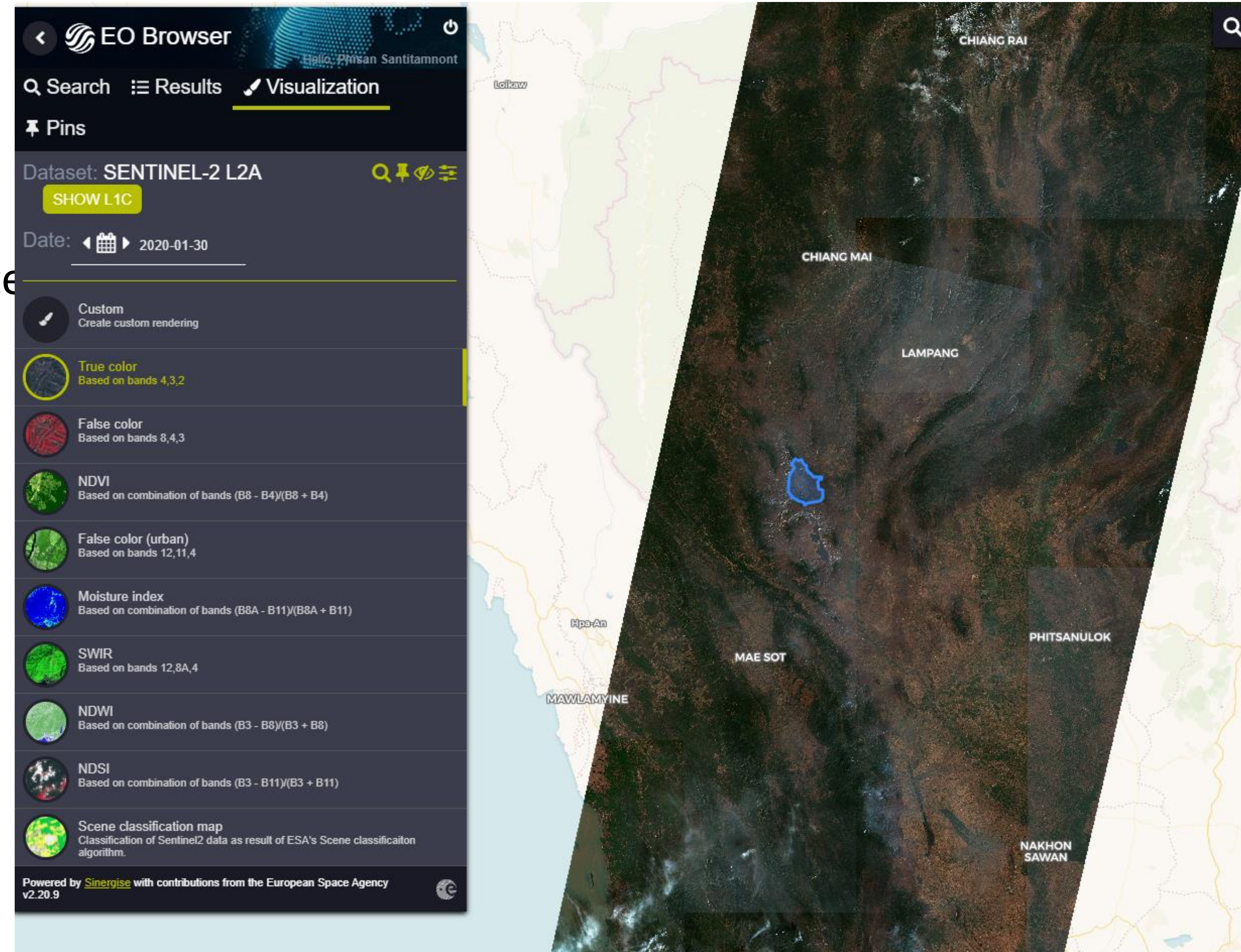
Exercise 3/4

Ex.3 Optical RS using S2A/S2B via Cloud



Composite

- Band Composite
- Band Ratio



จงวิเคราะห์ว่า Analysis Scheme เหมาะกับการประยุกต์ใช้อะไร

- True Color
- False Color
- NDVI
- NDWI
- SWIR
- .....

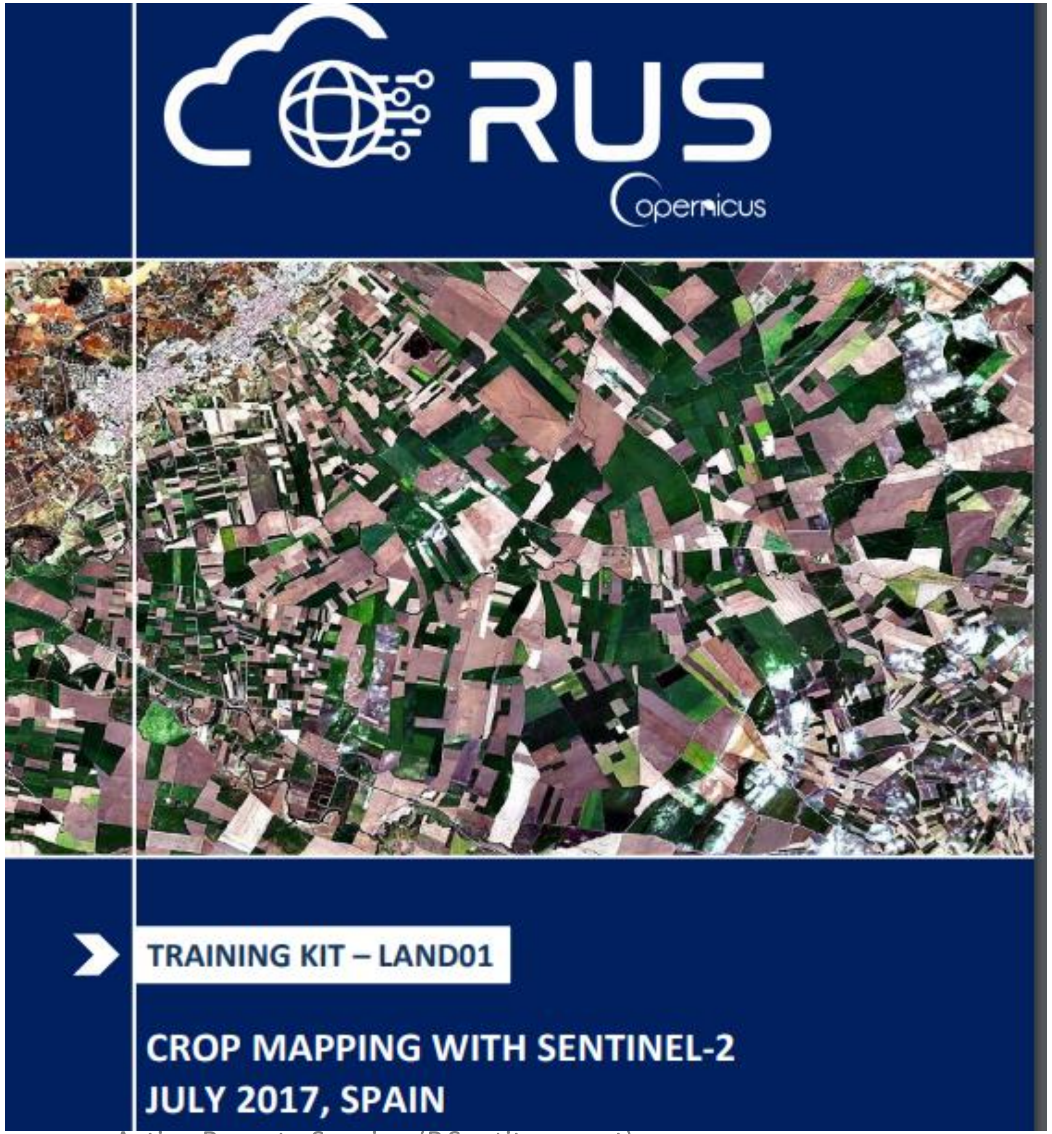
จงวิเคราะห์ สถานการณ์ไฟป่า จากภาพ SWIR/Sentinel2

- ให้ประมวลผล WildFire และ BurnScar ในเดือน มกราคม 2553 ในพื้นที่ “ลุ่มน้ำก้อ” ภายในพื้นที่อุทยานแห่งชาติแม่ปิง เหนือเขื่อนภูมิพล
- ให้เรียกใช้ข้อมูล HotSpot VIIRS เปรียบเทียบด้วย
- เปรียบเทียบภาพทั้งสองกลุ่ม จากนี้ ให้เขียน เวกเตอร์ และ ทิศทางของไฟ ด้วยมือ

- จากนั้นให้ขอ “custom script” ผ่าน Sentinel EO Browser ว่า ช่วยให้มองเห็นไฟป่า ชัดขึ้นอย่างไร
- Visualizing Wildfires and Burn Scars with the Sentinel Hub EO Browser V2

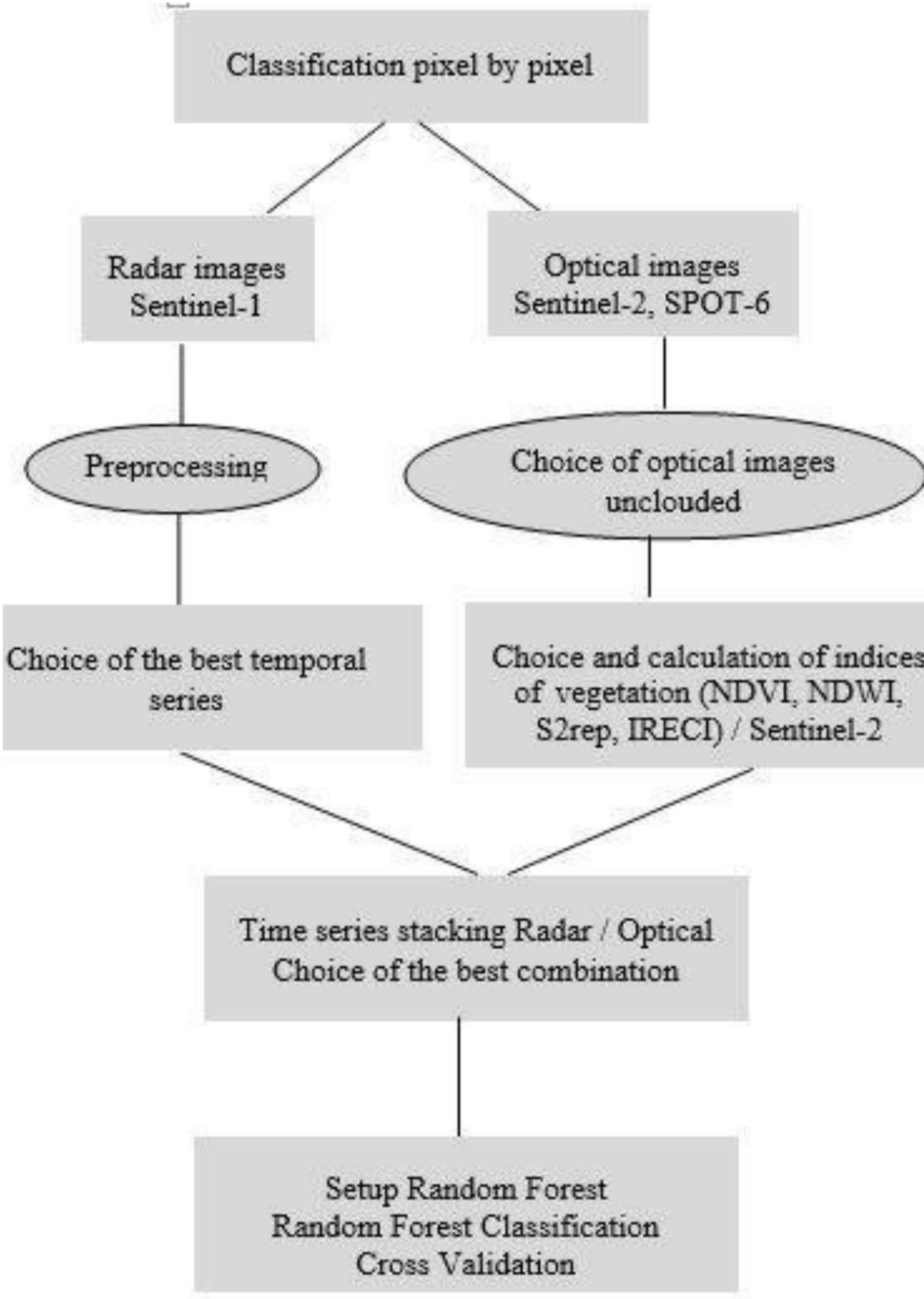
- https://pierre-markuse.net/2018/04/30/visualizing-wildfires-burn-scars-sentinel-hub-eo-browser/

Ex.4 LandUse Mapping using Machine Learning (RF)



RF flexibility …

Land cover classes	Accuracy (%) SAR+NDVI+NWDI	Accuracy (%) SAR + S2rep + IRECI	Accuracy (%) SAR + Optical
Summer crops	84	74	75
No vegetation	95	92	98
Forest/ undergrowth	81	85	84
Water	100	100	100
Grassland	83	87	73
Moors / Lawns	61	66	71
Winter crops	77	71	61
Sand dunes	-	-	-
Kappa index	**0.91**	**0.89**	**0.91**



Random Forest Classification

- Raster -> Classification -> Supervised Classification -> Random Forest Classifier
- Random Forest algorithm is a machine learningtechnique
- opposition to parametric classifiers (e.g. Maximum Likelihood),
- Learning the relationship between the training and the
- response dataset.
- The Random Forest classifier is an aggregated model, which means it uses the output from different models (trees) to calculate the response variable.

Increasing accuracy … more machine learning (group work ….)

- 2017_06_11 → S2A_MSIL2A_20170611T110621_N0205_R137_T30STG_20170611T111012
  2017_06_21 → S2A_MSIL2A_20170621T110651_N0205_R137_T30STG_20170621T111222
  2017_07_01 → S2A_MSIL2A_20170701T111051_N0205_R137_T30STG_20170701T111746
  2017_07_11 → S2A_MSIL2A_20170711T110651_N0205_R137_T30STG_20170711T111223
  2017_07_21 → S2A_MSIL2A_20170721T110621_N0205_R137_T30STG_20170721T112025
  2017_07_31 → S2A_MSIL2A_20170731T110651_N0205_R137_T30STG_20170731T111220

การประมวลผล Processing Scheme

เครื่องมือประมวลผล

- Online Suite: Sentinel Hub/Synergize , Google Earth Engine (GEE)
- Interactive ESA SNAP ver. 8.0, ENVI
- Script off-line
  - GDAL utility, Python , rasterio
  - SNAP GPF (Graph Processing FrameWork)
    - Python os.command(…..
  - Python API : python/snappy

MIST_Ex.5 Script for Cloud RS Application

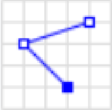
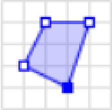
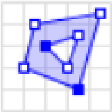
- Sentinelsat Python package allows users to make OGC (WMS and WCS) web requests to download and process satellite images within your Python scripts. It supports Sentinel-2 L1C and L2A, Sentinel-1, Landsat 8, MODIS and DEM data source.

- py -3 -m pip install pandas
- py -3 -m pip install geopandas
- py -3 -m pip install sentinelsat

Well-known Text : WKT

- originally defined by the Open Geospatial Consortium (OGC) , ISO/IEC 13249-3:2016
- https://arthur-e.github.io/Wicket/sandbox-gmaps3.html

Geometry primitives (2D)

Type	Examples	
Point		POINT (30 10)
LineString		LINESTRING (30 10, 10 30, 40 40)
Polygon		POLYGON ((30 10, 40 40, 20 40, 10 20, 30 10))
		POLYGON ((35 10, 45 45, 15 40, 10 20, 35 10), (20 30, 35 35, 30 20, 20 30))

Example 1

- user = 'user' ## change this!
- password = 'password' ## change this!

- api = SentinelAPI (user, password, 'https://scihub.copernicus.eu/dhus')

- footprint = “POINT(100 14)”
- print (footprint)

- products = api.query(footprint,
- date = ('20190225', '20190227'),
- platformname = 'Sentinel-2',
- processinglevel = 'Level-2A',
- cloudcoverpercentage = (0, 20))

- areas = api.to_geodataframe(products)
- areas.plot(column='uuid', cmap=None)

Download Data

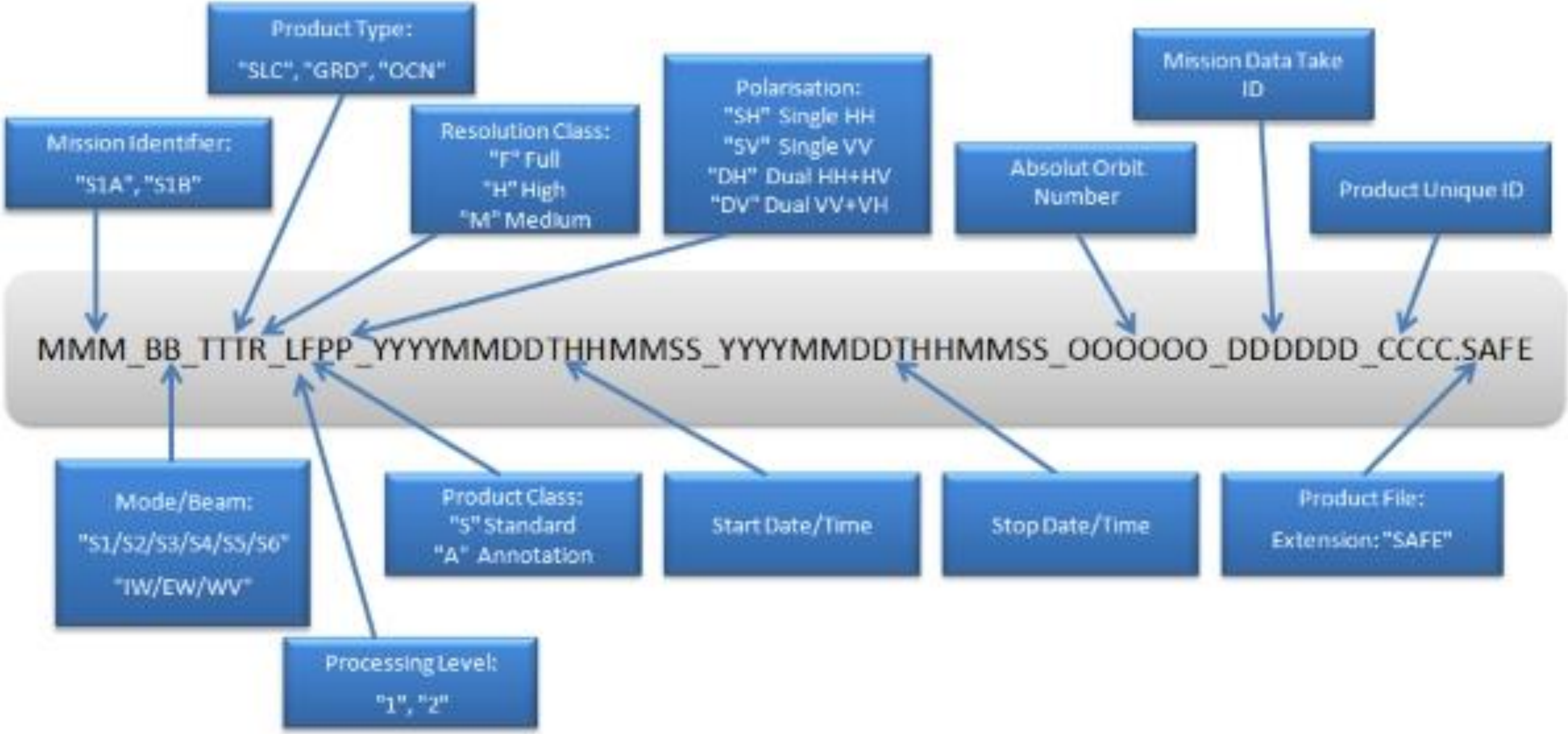
```
from sentinelsat import SentinelAPI, read_geojson, geojson_to_wkt

api = SentinelAPI('user', 'password')
footprint = geojson_to_wkt(read_geojson('/path/to/search_polygon.geojson'))
products = api.query(footprint,
                     producttype='SLC',
                     orbitdirection='ASCENDING')
api.download_all(products)
```

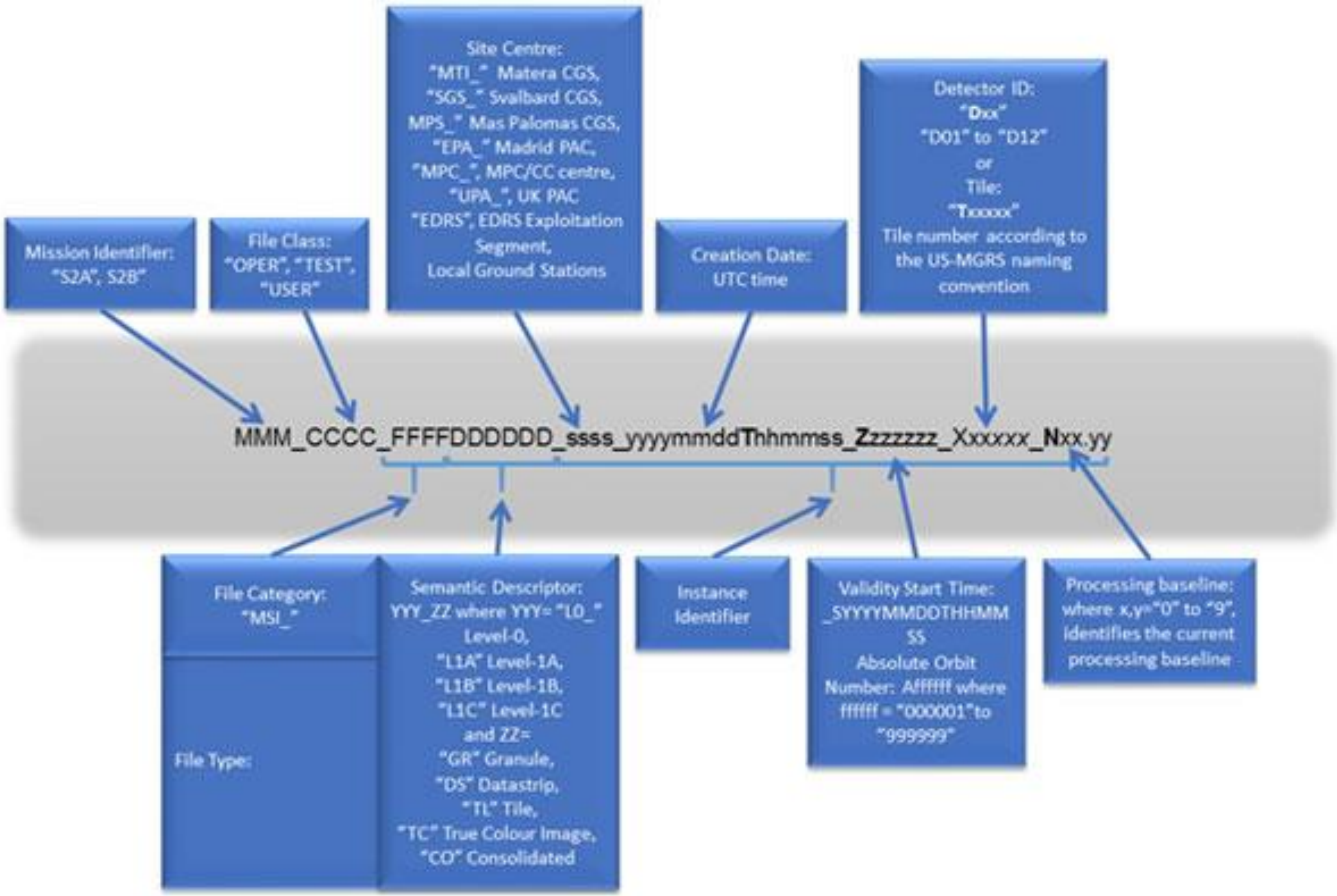
OpenSearch

```
>>> api.query(date=('NOW-8HOURS', 'NOW'), producttype='SLC')
OrderedDict([('04548172-c64a-418f-8e83-7a4d148adf1e',
              {'acquisitiontype': 'NOMINAL',
               'beginposition': datetime.datetime(2017, 4, 25, 15, 56, 12, 814000),
               'endposition': datetime.datetime(2017, 4, 25, 15, 56, 39, 758000),
               'filename': 'S1A_IW_SLC__1SDV_20170425T155612_20170425T155639_016302_
↪01AF91_46FF.SAFE',
```

S1A S1B Naming Convention



S2A S2B Naming Convention



A product's availability can be checked with a regular OData query by evaluating the `Online` property value.

```
product_info = api.get_product_odata(<product_id>)

if product_info['Online']:
    print('Product {} is online. Starting download.'.format(<product_id>))
    api.download(<product_id>)
else:
    print('Product {} is not online.'.format(<product_id>))
```

When trying to download an offline product with *download()* it will trigger its retrieval from the LTA.

Given a list of offline and online products, *download_all()* will download online products, while concurrently triggering the retrieval of offline products from the LTA. Offline products that become online while downloading will be added to the download queue. *download_all()* terminates when the download queue is empty, even if not all products were retrieved from the LTA. We suggest repeatedly calling *download_all()* to download all products, either manually or using a third-party library, e.g. tenacity.

```
from sentinelsat import SentinelAPI
import tenacity

api = SentinelAPI('user', 'password')

@tenacity.retry(stop=tenacity.stop_after_attempt(3), wait=tenacity.wait_fixed(3600))
def download_all(*args, **kwargs):
    return api.download_all(*args, **kwargs)

downloaded, triggered, failed = download_all(<product_ids>)
```

พื้นที่ทดสอบ เหมืองเปิด “แม่เหมาะ”

- **ในรอบ 3ปี ผ่านมา**

- ก) มีข้อมูลเรดาร์ S1A , S1B ชนิด GRD รวมแล้วกี่ภาพ
- พล็อตด้วย geodatframe ให้ดู
- แยกเป็น ascending, desending orbit อย่างไร
- โดยเฉลี่ยมีภาพทุกๆ กี่วัน
- ข) มีข้อมูลมัลติสเปคตรัม S2A , S2B ชนิด L1C กี่ภาพ
- กำหนดปริมาณเมฆเป็น 10%
- พล็อตด้วย geodataframe ให้ดู
- แยกเป็น ascending, descending orbit อย่างไร
- ภาพ MSI มีสม่ำเสมอ หรือไม่ อย่างไร

ขอบคุณครับ