

# Geospatial Data Analysis with Python

## FOSS4G Thailand 2019

*Mr. Chaipat Nengcomma*  
Ayutthaya-GIS Co., Ltd.



### Ayutthaya GIS Consultant

Creating Business Solutions Through  
Spatial Data Analysis

## Chaipat nengcomma



- 15+ years professional experience developing Geographic Information System (GIS)
- Spatial Data Scientist , with deep background in GIS and Advance Remote sensing technology
- Skilled in Python (Programming Language), Database Management ,
- Experience developing web application project
- solving complex business problems using data science by applying advanced predictive modelling techniques, Machine Learning and AI.
- Finished *Deep learning specialization certification* , five-course about Deep Learning from Deeplearning.ai
- Finished *Data Science Math Skills certification* from Duke University

## Goals & Objectives

## Goals & Objectives

- แนะนำภาษา Python กับการวิเคราะห์ข้อมูล Geospatial Data\*\*\*
- นำเสนอภาพรวม การต่อ�อดพัฒนาระบบ
- ทำความรู้จัก เครื่องมือ/ไลบรารี ที่นักพัฒนาสามารถนำไปใช้งานได้
- ทำความรู้จัก แหล่งข้อมูล และวิธีการเข้าถึง เพื่อนำไปใช้งาน

Mr. Chaipat @Ayutthaya GIS

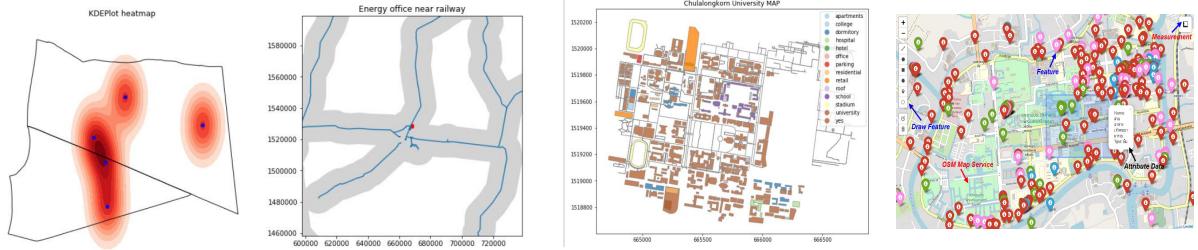
8

## Workshop Content

9

# Content

1. Geospatial Data Analysis
2. Web Mapping Application
3. Geospatial Data Service {OSM, WFS, WMS, Geocode}



Mr. Chaipat @Ayutthaya GIS

10

## Part I : Introduction

---

11

# What Can I Do With Python + Open Source GIS ?? ตัวอย่างการนำไปใช้ของที่ Lab

12

*Drone Mapping*

13

## Smart Drone Mapping

- ระบบประมวลผลภาพถ่ายจากโดรนแบบอัตโนมัติ
- ความรวดเร็วในการสำรวจและจัดทำแผนที่รายละเอียดสูง
- รองรับการประยุกต์ใช้งาน สำรวจเก็บข้อมูลที่ดิน, จัดการอาคาร และงานด้านการเกษตร

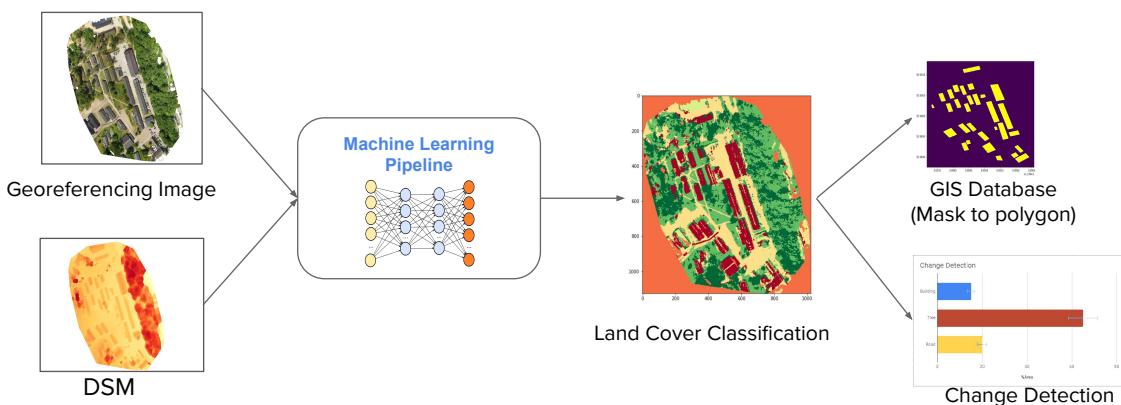


Mr. Chaipat @Ayutthaya GIS

14

## Smart Drone Mapping

- นำ AI มาใช้ชีวิเคราะห์ข้อมูลภาพถ่ายจากโดรน เพื่อการจำแนกภัณฑ์ในภาพ แบบอัตโนมัติ รองรับการผลิตแผนที่ (Mapping) และสารสนเทศเชิงปริภูมิ (GIS) แบบเร่งด่วน รองรับการเชื่อมต่อกับระบบ Map Service พิรุณเขื่อมต่อข้อมูลบน GIS Application



15

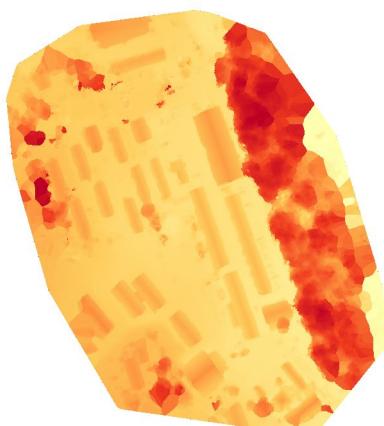
## Raw Data



## Processing

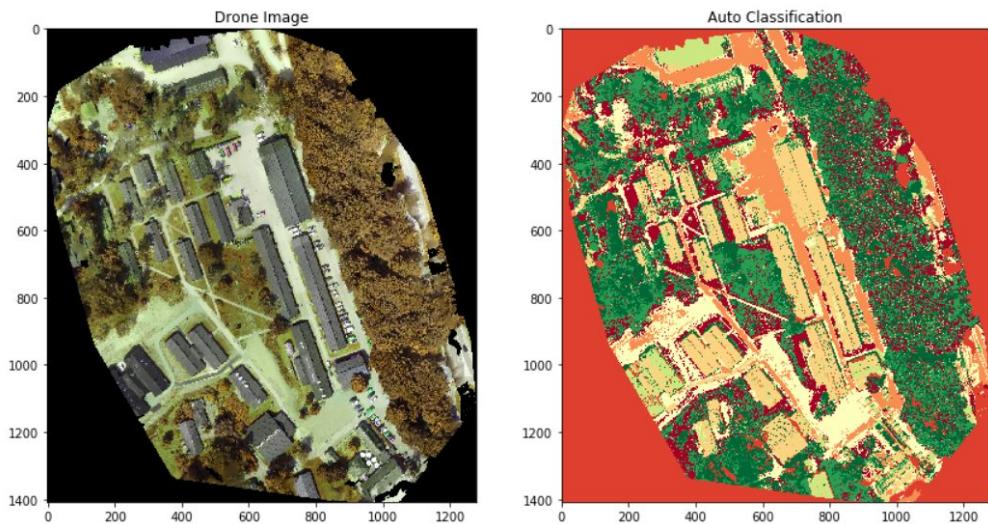


Ortho Image



DSM Image

## Auto Classification with AI



## Land Cover Classification

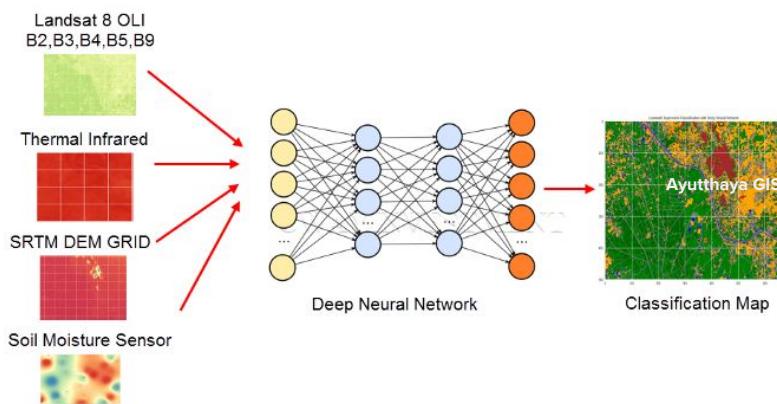
# Land Cover Classification

- Supervised Classification Multi-Sensor Data Integration Using Deep Learning
- ประมวลผลถ่ายดาวเทียมแบบ Supervised Classification โดยการพัฒนา complex model ที่วิเคราะห์ผลกับข้อมูลขนาดใหญ่ จากหลายแหล่ง แบบ Multi-Sensor มาทำการวิเคราะห์ร่วมกัน เพื่อสร้างข้อมูลผลลัพธ์การจำแนก ที่แม่นยำและมีประสิทธิภาพขึ้น

Mr. Chaipat @Ayutthaya GIS

20

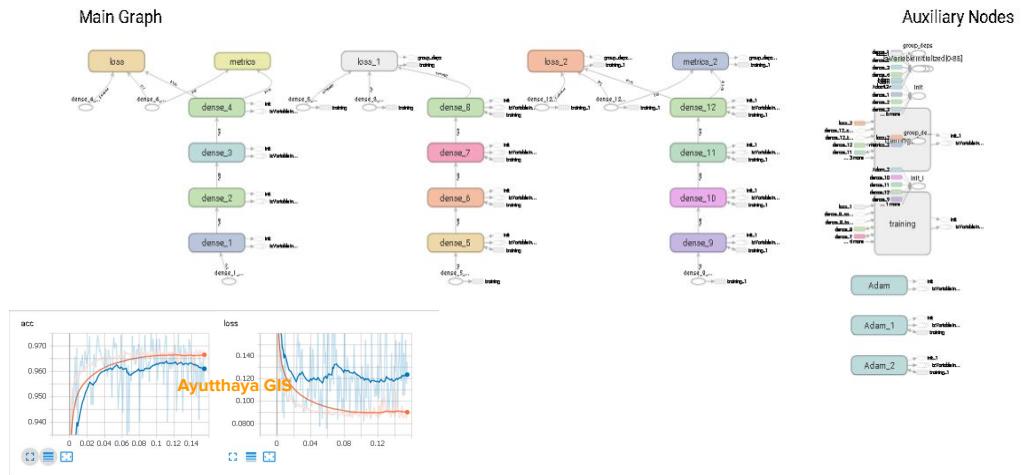
# Land Cover Classification



Mr. Chaipat @Ayutthaya GIS

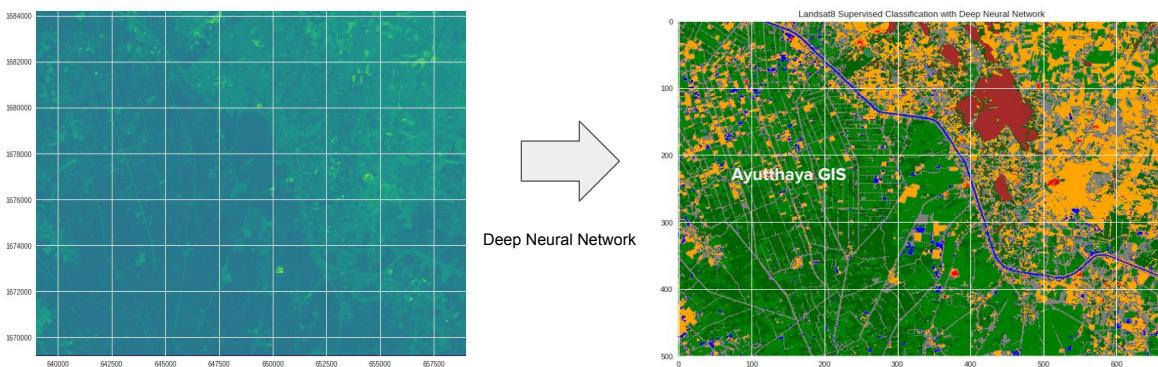
21

## Land Cover Classification



22

# Land Cover Classification

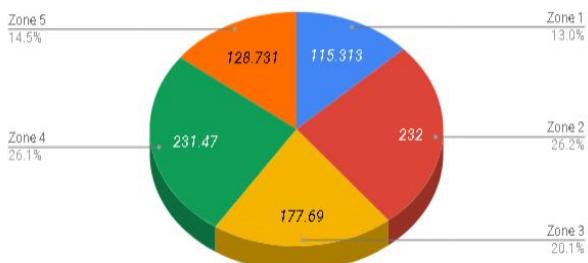


Mr. Chaipat @Ayutthaya GIS

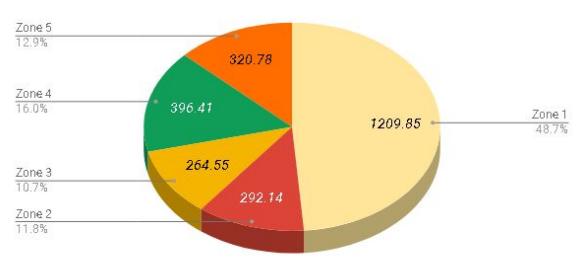
23

# Green Project

ขนาดพื้นที่ (ตร.กม)



ความหนาแน่นประชากรต่อพื้นที่

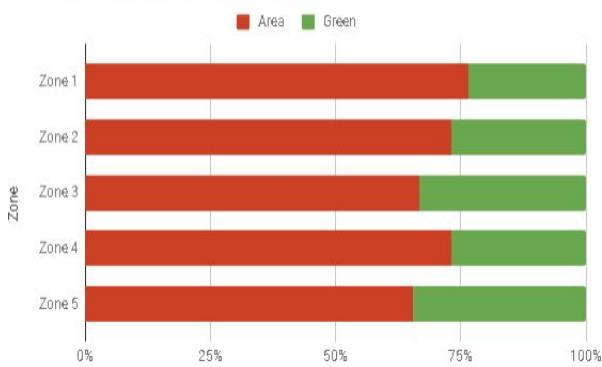


Mr. Chaipat @Ayutthaya GIS

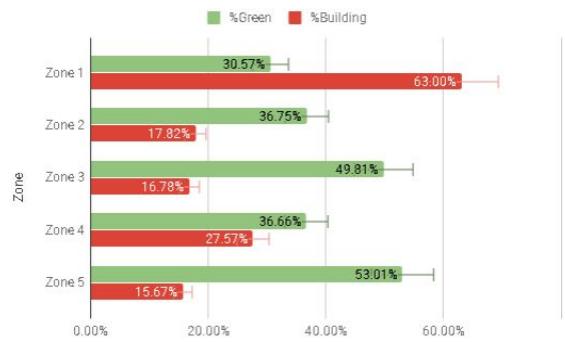
24

# Green Project

ปริมาณเพื้นที่สีเขียว หรือ ขนาดพื้นที่



ปริมาณเพื้นที่สีเขียว หรือ ปริมาณอาคาร สิ่งปลูกสร้าง



Mr. Chaipat @Ayutthaya GIS

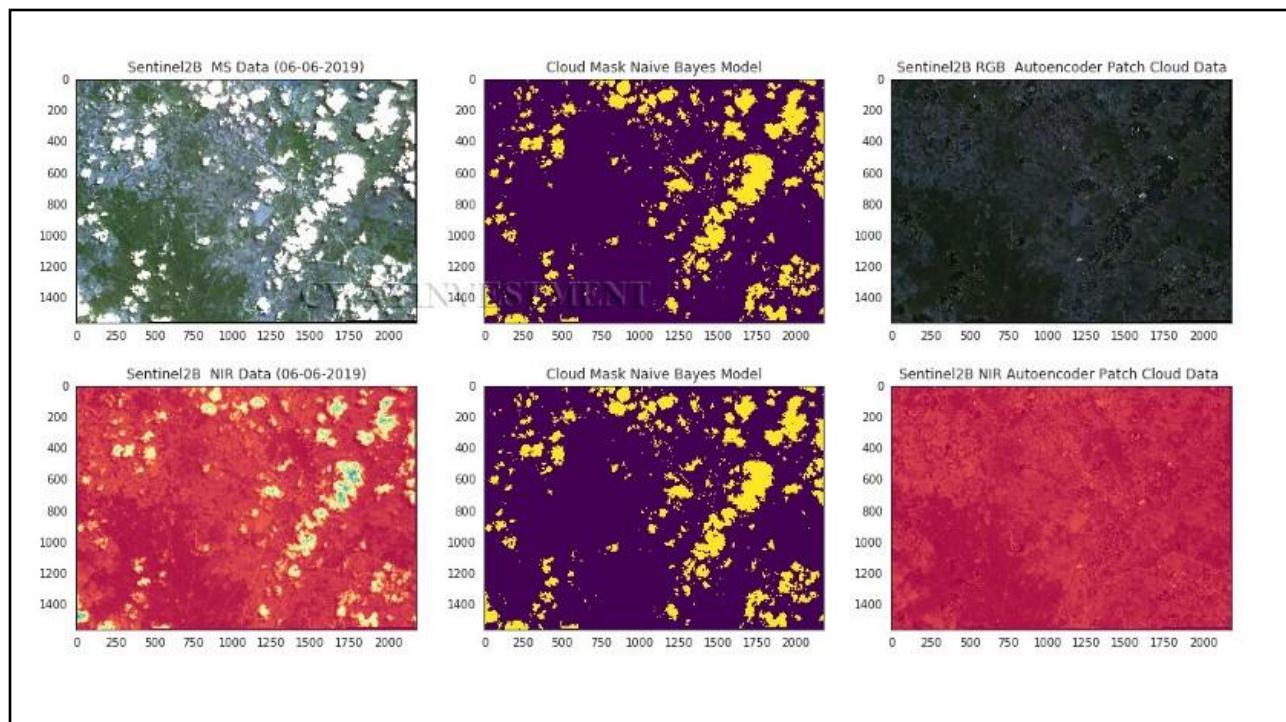
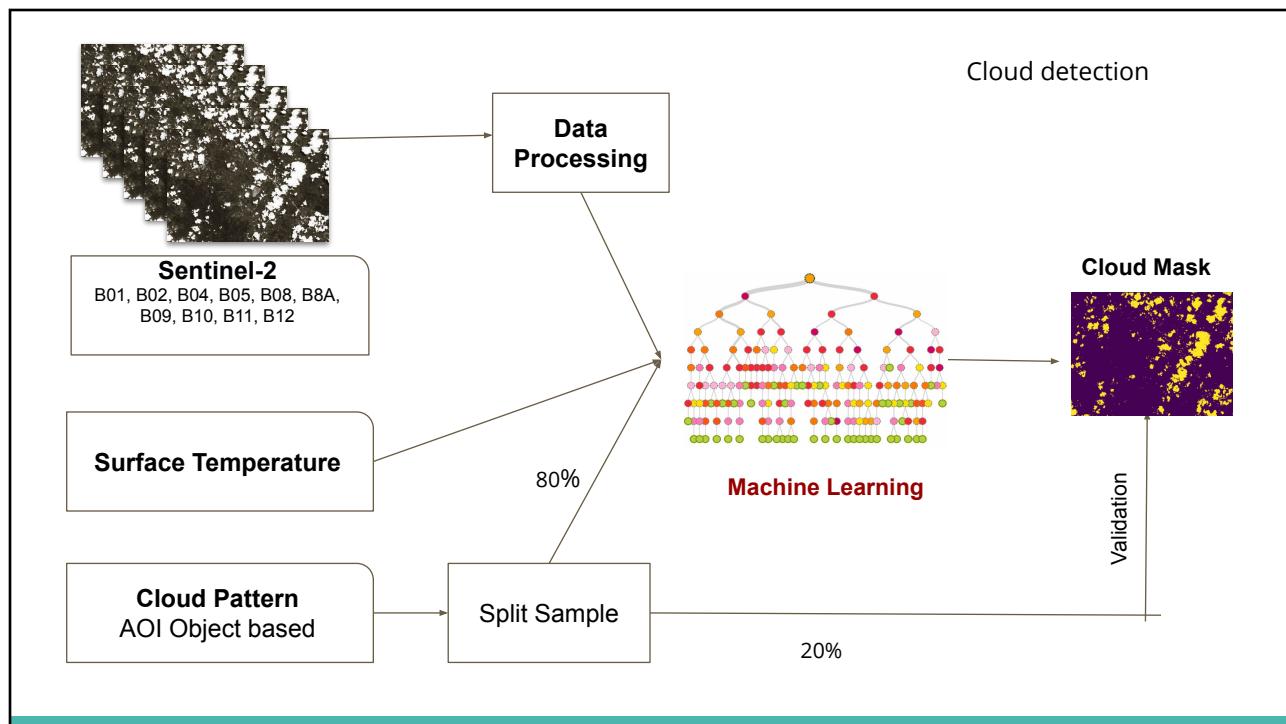
25

## Crop assessment and monitoring

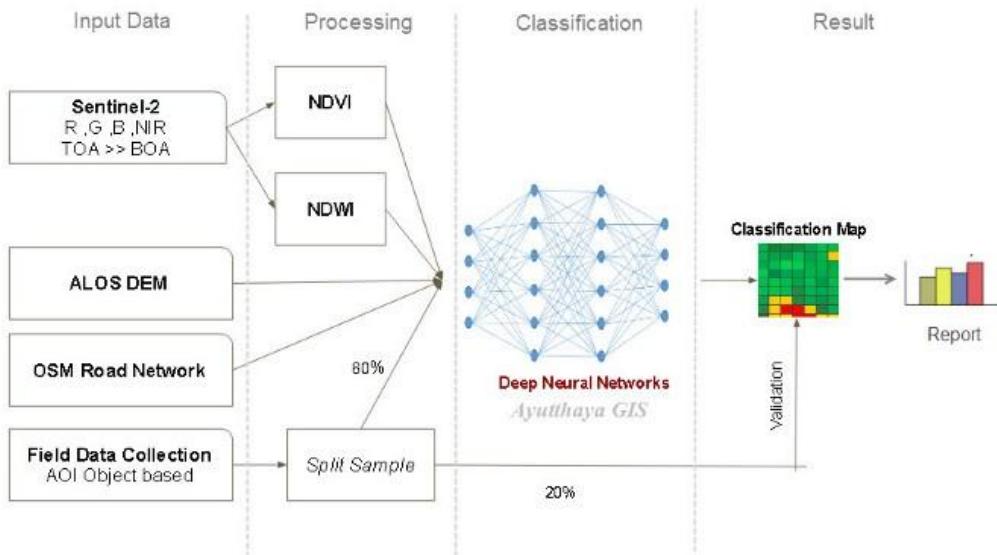
26

### Crop assessment and monitoring

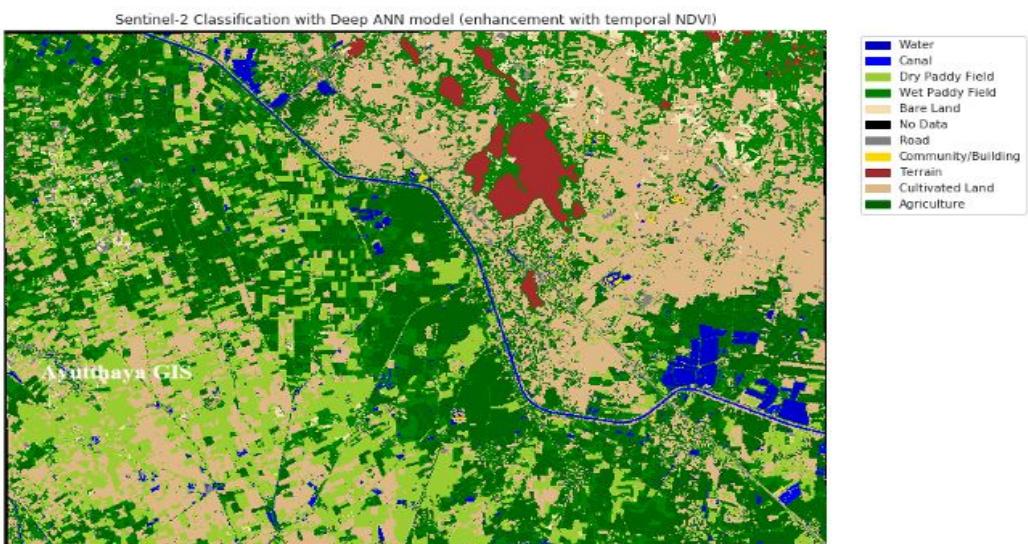
- Supervised Classification
- ทำระบบอัตโนมัติที่มาตอบโจทย์เรื่อง การติดตามการเพาะปลูก(Crop assessment & Crop monitoring)
- ข้อมูลภาพถ่ายดาวเทียม sentinel-2 ที่บันทึกการถ่ายภาพทุกๆสัปดาห์
- การพัฒนา AI มาช่วย โดยอาศัย Deep Learning ที่ปัจจุบันกลายเป็น state-of-the-art เพื่อประมวลผลสักดิ้น insight กับข้อมูลขนาดใหญ่



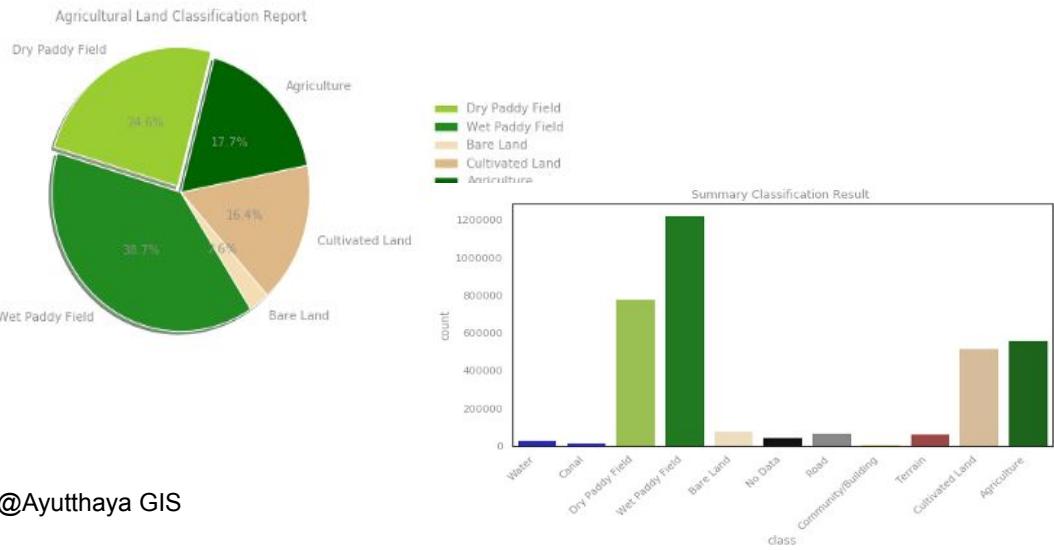
# Land Cover Classification



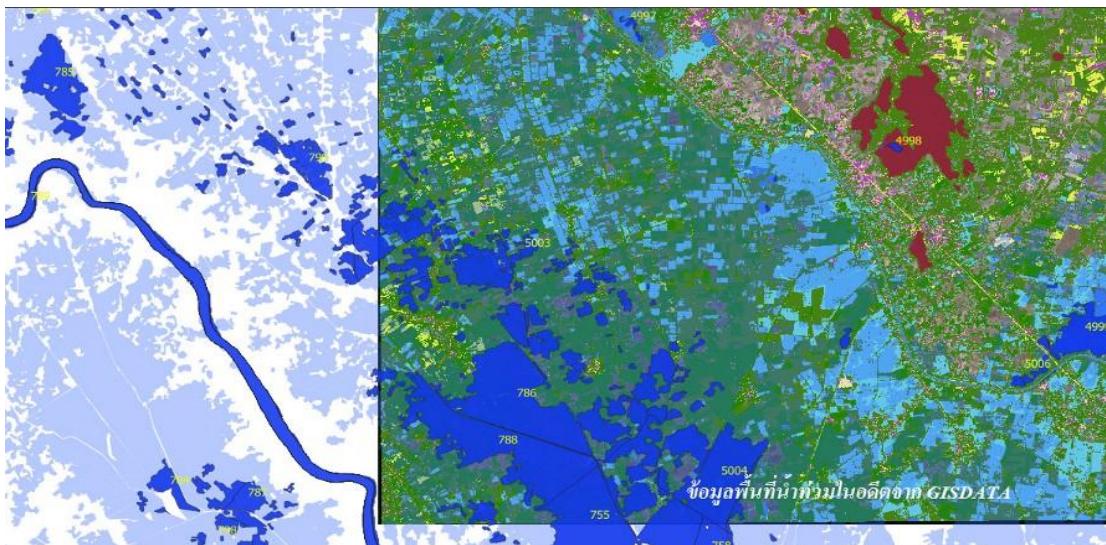
# Land Cover Classification



# Crop assessment



## Risk Management & Crop monitoring



# Smart Farm

34

## Why Smart Farm

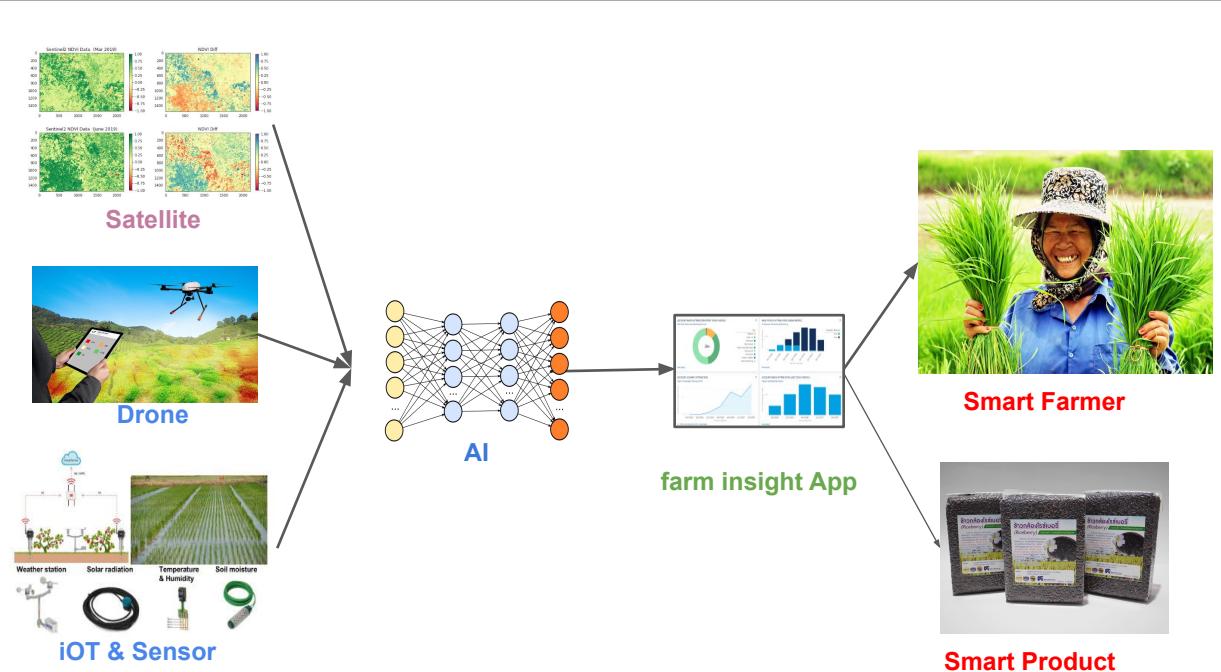
- ผลกระทบจากสภาพอากาศที่แปรปรวน ส่งผลต่อปริมาณและคุณภาพผลผลิต
- ต้นทุนการผลิต ที่เพิ่มสูงขึ้น
- การแข่งขันการตลาดที่สูง ภาวะราคาผลผลิตที่ไม่แน่นอน
- ขาดแรงงานที่ชำนาญ ปริมาณเกษตรกรลดลงจาก การเปลี่ยนแปลงของโครงสร้างสังคม



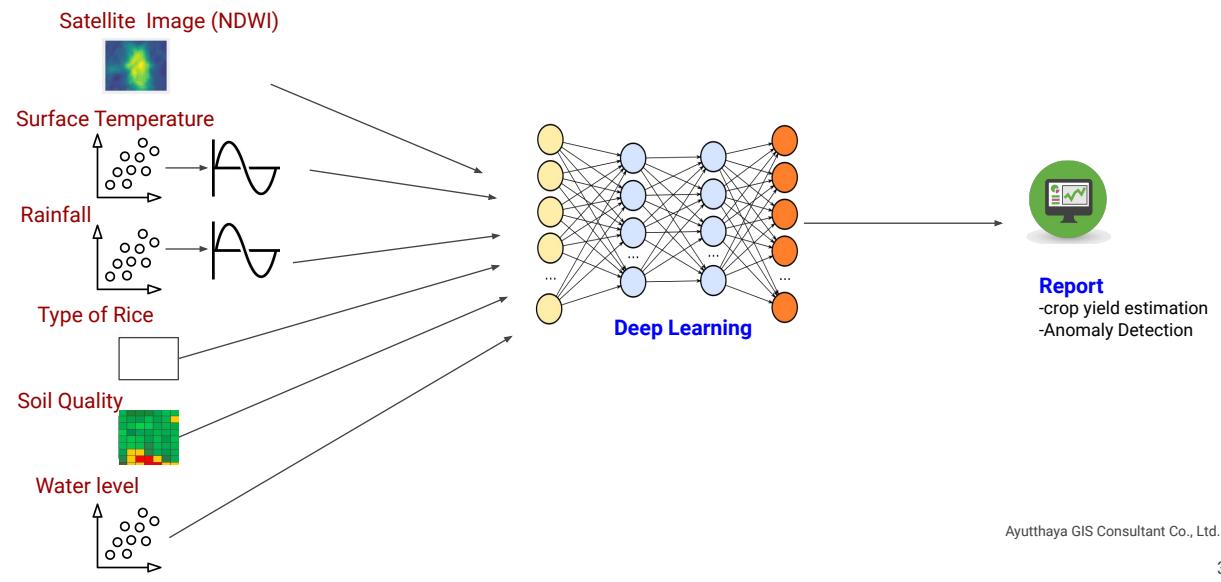
# Smart Farm

- การนำเทคโนโลยีมาใช้เพื่อบริหารจัดการและพัฒนาระบวนการผลิต
- เพื่อเพิ่มประสิทธิภาพ ลดต้นทุน เพิ่มปริมาณและคุณภาพผลผลิต และลดความเสี่ยงจาก การระบาดของศัตรูพืชและภัยธรรมชาติ
- Smart monitor == Remote sensing , Drone , Sensor , GPS/GNSS
- Smart management == GIS + Data analysis
- Smart planning == AI >> Decision Support System (DSS)

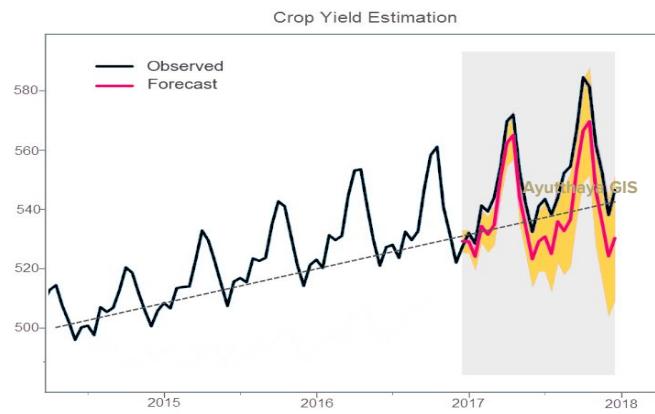
Ayutthaya GIS Consultant Co., Ltd.



# Crop Yield Estimation (Spatial Temporal)



## Crop Yield Estimation

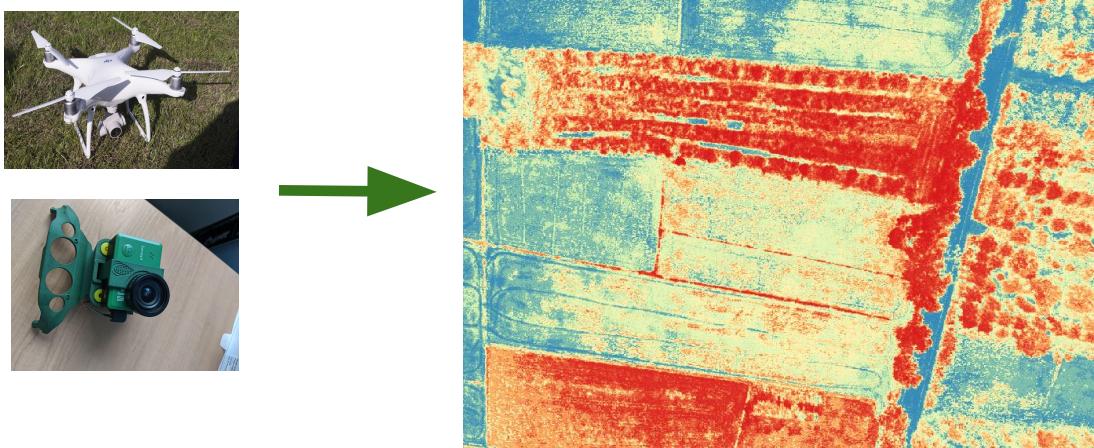


# Crop Health Monitor



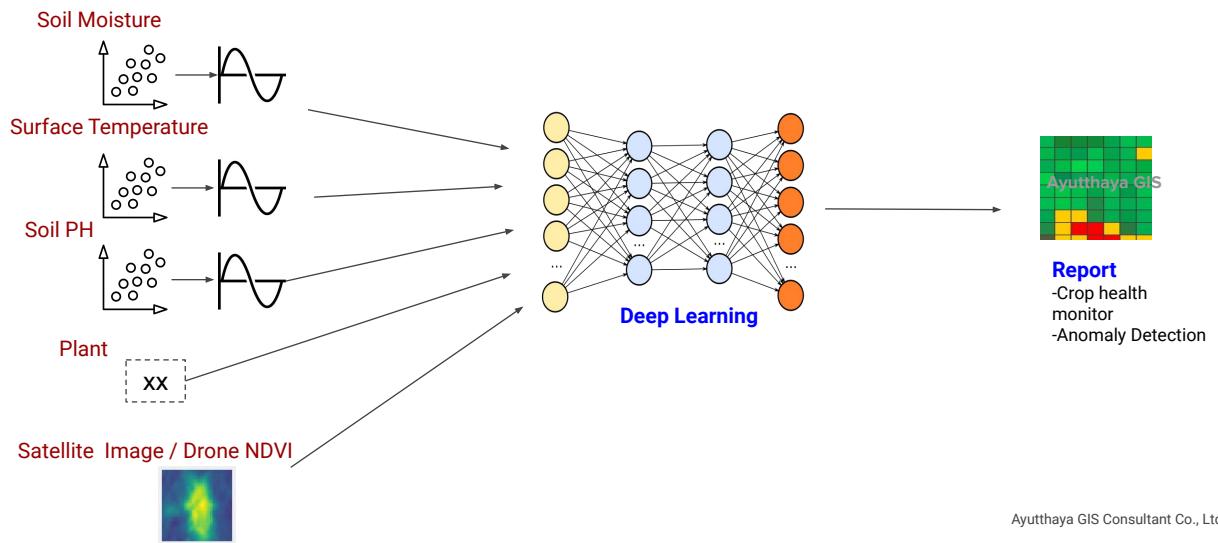
Ayutthaya GIS Consultant Co., Ltd.

# Crop Health Monitor

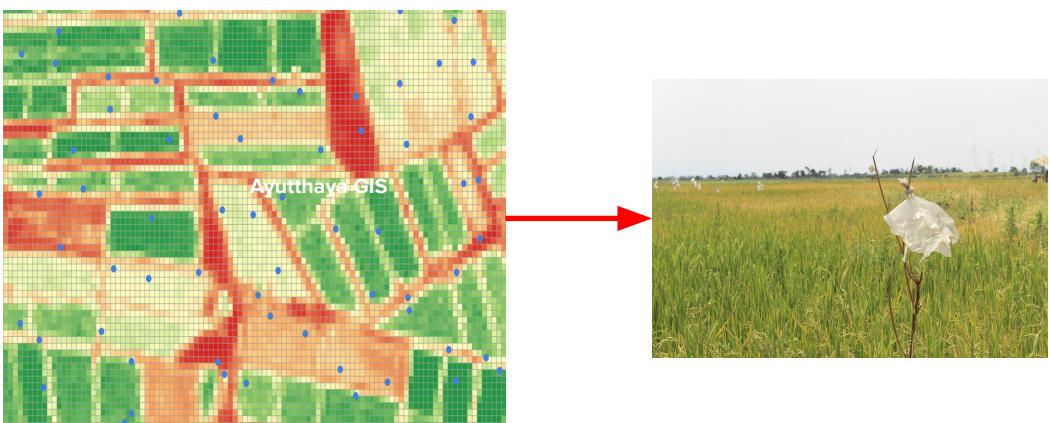


Ayutthaya GIS Lab & LING GIS Team

# Crop Health Monitor



# Crop Health Monitor

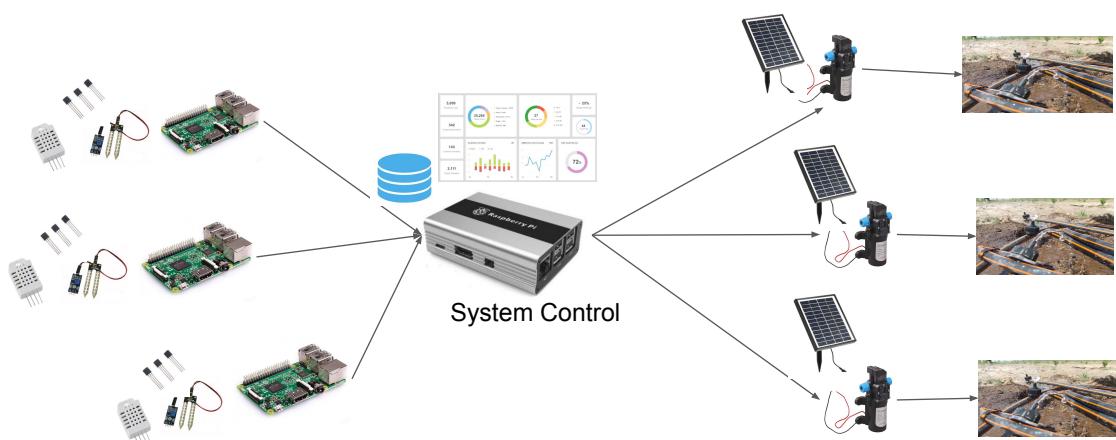


Ayutthaya GIS Consultant Co., Ltd.

# Automatic Plant Watering System

44

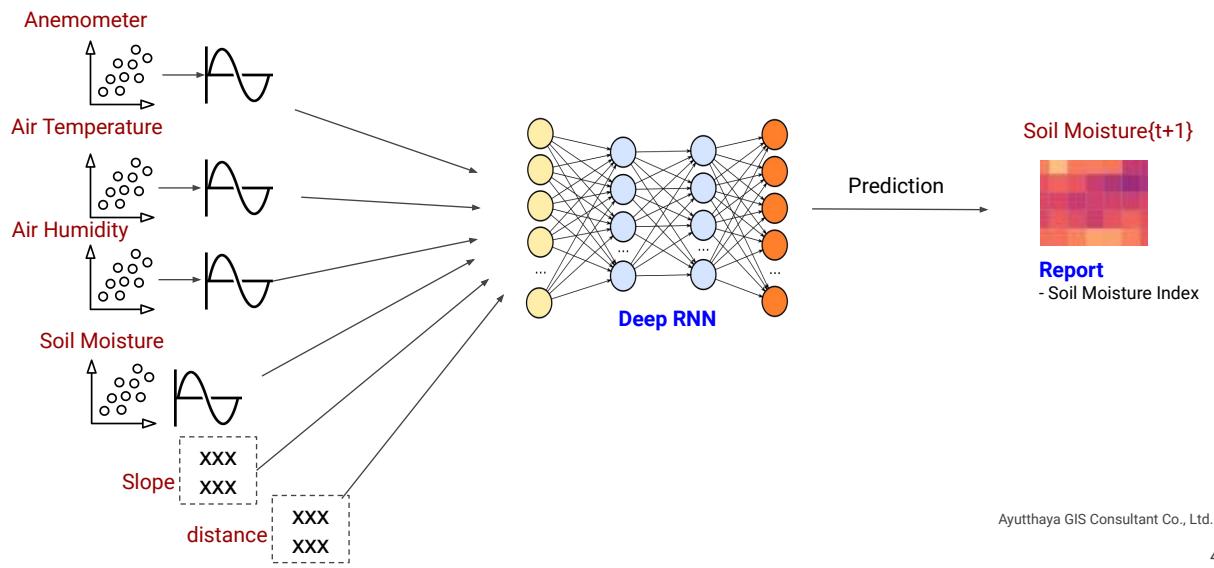
APWS



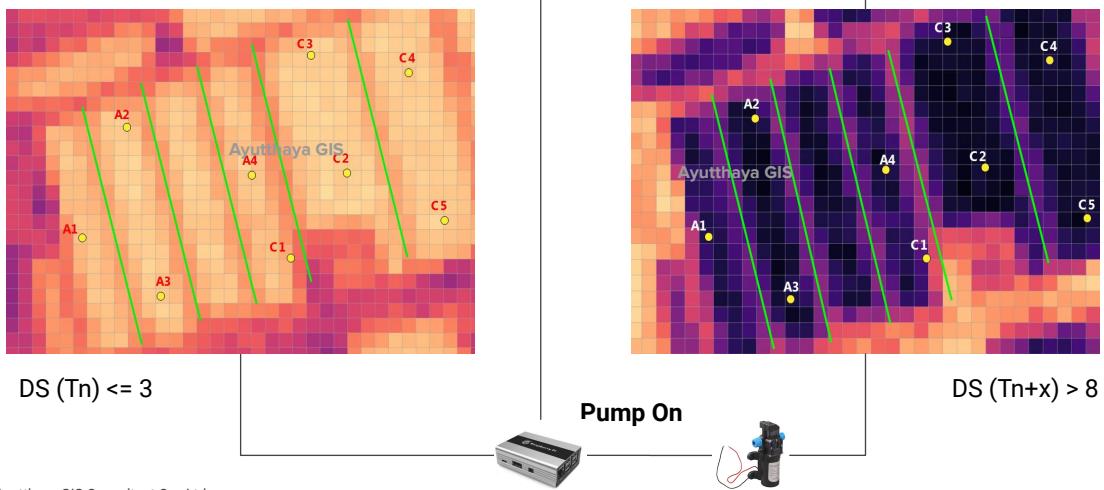
Ayutthaya GIS Consultant Co., Ltd.

45

# APWS

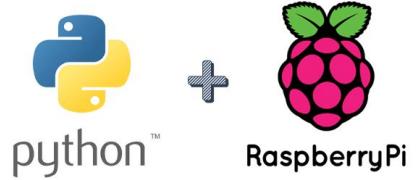


# APWS

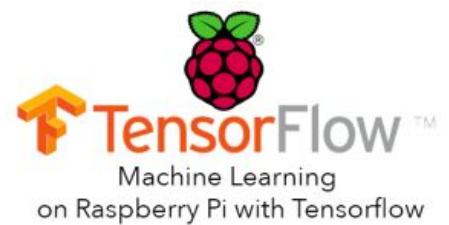


# APWS

- Raspbian
- Python , Tensorflow + Keras , Scikit learn
- Postgresql
- Flask



```
pi@raspberrypi:~ 
File Edit Tabs Help
pi@raspberrypi:~$ curl -O https://files.pythonhosted.org/reqs/666157500637faaeadaa95b04fffe74e12/setup tools-40.2.0-py2.py3-none-any.whl
pi@raspberrypi:~$ pip install --no-cache-dir --upgrade --force-reinstall .
100% [██████████] 573KB 350kB/s
Installing Collected packages: python-dateutil, setuptools, kiwisolver, pyparsing, matplotlib, numpy, tensorflow
Found existing installation: python-dateutil 2.5.3
    Not uninstalling python-dateutil at /usr/lib/python3/dist-packages, outside environment
Found existing installation: setuptools 33.1.1
    Not uninstalling setuptools at /usr/lib/python3/dist-packages, outside environment
Found existing installation: pyparsing 2.1.10
    Not uninstalling pyparsing at /usr/lib/python3/dist-packages, outside environment
Found existing installation: matplotlib 2.0.0
    Not uninstalling matplotlib at /usr/lib/python3/dist-packages, outside environment
Successfully installed kiwisolver-1.0.1 matplotlib-3.0.0rc2 pyparsing-2.2.0 python-dateutil-2.7.3 setuptools-40.2.0
pi@raspberrypi:~$ python3 -c 'import keras; print(keras.__version__)' # for PyCharm
2.2.2
pi@raspberrypi:~$ 
```



# Introduction

# Introduction

- Python เป็นภาษาโปรแกรมคอมพิวเตอร์ระดับสูง มากความสามารถ รองรับการทำงานข้ามแพลต์ฟอร์มระบบปฏิบัติการ, รองรับการทำงานหลากหลายประเภท
- GIS คือระบบสารสนเทศเชิงพื้นที่ + การบริหารจัดการข้อมูลทำให้เกิดประโยชน์
- Open Source GIS software คือ ซอฟต์แวร์ พัฒนาระบบ GIS แบบรหัสเปิด

Mr. Chaipat @Ayutthaya GIS

50

# Introduction

- Open Source Geospatial Foundation (OSGeo) มูลนิธิ Geospatial Open Source เป็นองค์กรไม่แสวงหาผลกำไร
- ภารกิจคือการสนับสนุนและส่งเสริมการพัฒนาความร่วมมือของเทคโนโลยีและข้อมูลภูมิสารสนเทศแบบเปิด



Mr. Chaipat @Ayutthaya GIS

51

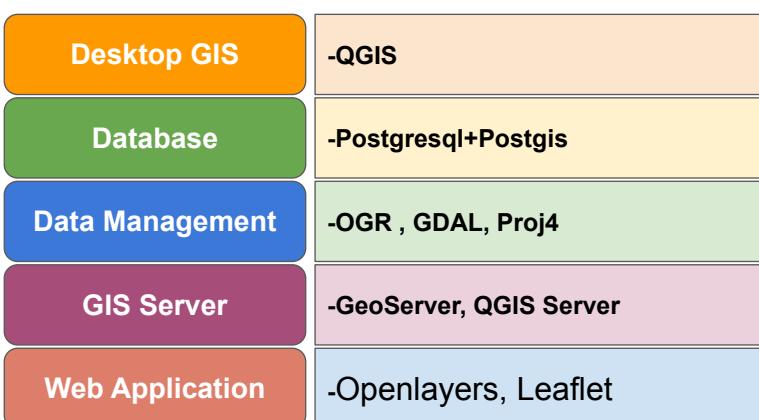
# Open Source GIS

<b>Desktop GIS</b>	QGIS, GRASS GIS
<b>Data Management API</b>	GDAL , OGR , Proj4
<b>Spatial Database</b>	Postgresql+Postgis , Spatial lite (SQLite),
<b>GIS Server</b>	GeoServer , MapServer, Qgis Server
<b>Web Application Framework</b>	Openlayers, Leaflet

Mr. Chaipat @Ayutthaya GIS

52

# Open Source GIS



53

## Open Source GIS



Mr. Chaipat @Ayutthaya GIS

54

## Python & GIS

- Python เป็นภาษาคอมพิวเตอร์ที่เป็น tools สำคัญในงานด้านการวิเคราะห์ข้อมูลเชิงพื้นที่ โดยเฉพาะลักษณะ กระบวนการทำงานแบบ อัตโนมัติ
- รองรับด้านการจัดการข้อมูล , การประมวลผล การแสดงผล รวมไปถึงความสามารถในการพัฒนา application ต่างๆ
- ภาษาพื้นฐานใน GIS Software เช่น QGIS , ArcGIS, GRASS ช่วยเพิ่มประสิทธิภาพในการทำงาน

Mr. Chaipat @Ayutthaya GIS

55

## Python & GIS

- Python เป็นเหมือนตัวเชื่อมประสาน การทำงานร่วมกันระหว่างนัก GIS กับ Developer
- มีการเติบโต และขยายของนักพัฒนา ทำให้มี opensource lib รองรับจำนวนมาก ส่วนหนึ่งต่อยอดจาก opensource gis เดิม รวมถึงการพัฒนาโมเดลใหม่ๆในลักษณะ GIS programming stack

Mr. Chaipat @Ayutthaya GIS

56

## Python & GIS

- ขยายขีดความสามารถในการเข้าถึงและวิเคราะห์ข้อมูล spatial data ในระบบงานที่ซับซ้อน
- งานด้าน Data science ,
- นำข้อมูลเชิงพื้นที่ มาสู่กระบวนการการพัฒนา AI (Machine Learning , Deep Learning)
- ธุรกิจ / บริการ ที่ต่อยอดจากข้อมูลเชิงพื้นที่

Mr. Chaipat @Ayutthaya GIS

57

# Spatial Data science

58

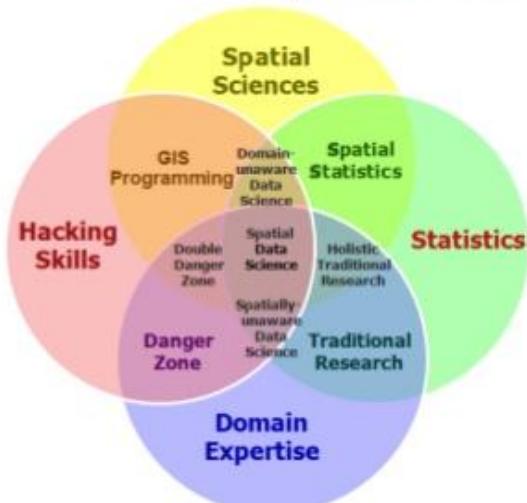
## Spatial Data science

- Data science กระบวนการทางวิทยาศาสตร์เพื่อวิเคราะห์ เจาะลึก ทำความเข้าใจ เพื่อสกัดนำมูลค่าจากข้อมูลขนาดใหญ่ ข้อมูลที่มีโครงสร้างซับซ้อน
- Spatial Data = ข้อมูลเชิงปริภูมิ / ข้อมูลภูมิสารสนเทศ / ข้อมูลเชิงพื้นที่
- Key คือ ข้อมูลที่อ้างอิงกับตำแหน่งบนผิวโลก

Mr. Chaipat @Ayutthaya GIS

59

# Spatial Data Science



All they are combined for **data analysis** in order to ...

Support a better decision making

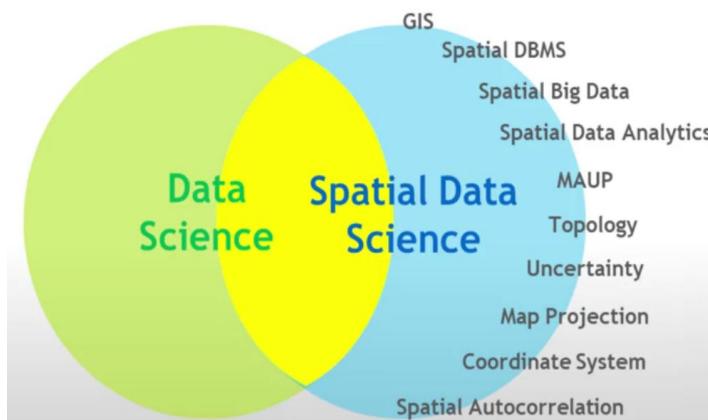
"The key word in data science is not data; it is science"  
Jeff Leek. Data Science Specialization. Coursera.

Alí Santacruz ,amsantac.co

5

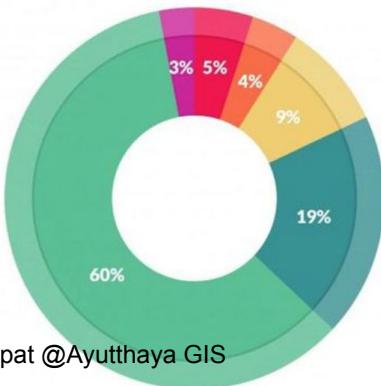
60

## Spatial Data science



# Data science

**Data preparation** accounts for about 80% of the work of data scientists



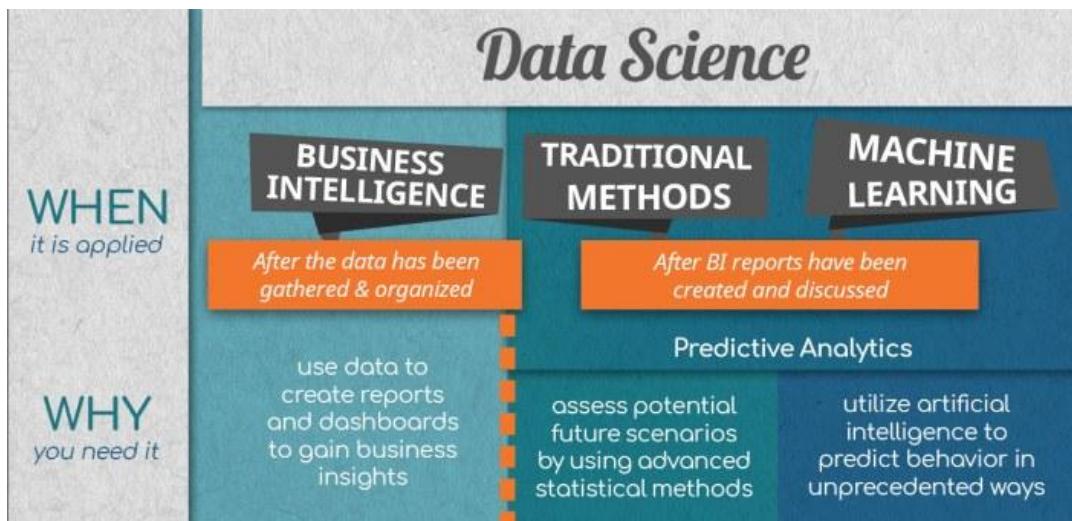
Mr. Chaipat @Ayutthaya GIS

What data scientists spend the most time doing

- Building training sets: 3%
- Cleaning and organizing data: 60%
- Collecting data sets; 19%
- Mining data for patterns: 9%
- Refining algorithms: 4%
- Other: 5%

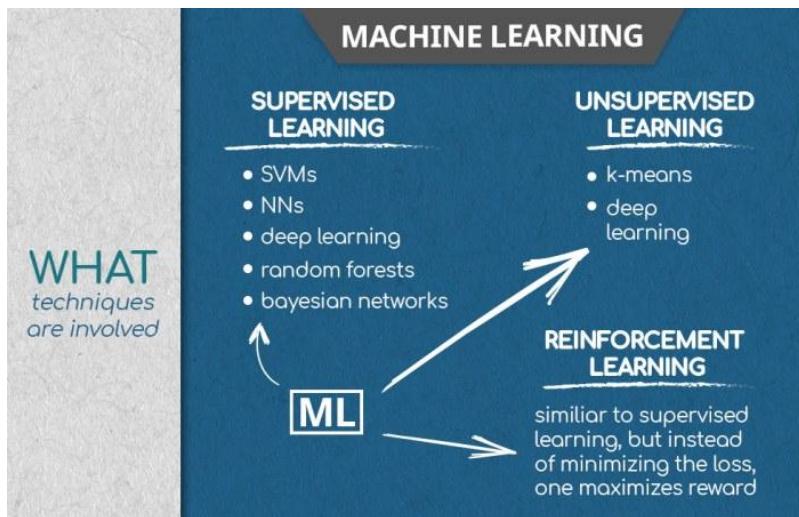
62

# Data science



63

# Machine Learning



64

## Part II : Development

---

65

# Python Geospatial Development Stack

66

## Python Geospatial Development Stack

Layer	Package Name	Description	URL
<i>Spatial Data IO</i>	gdal	Interface to Geospatial data abstraction library	<a href="https://pypi.python.org/pypi/GDAL/">https://pypi.python.org/pypi/GDAL/</a> (link is external)
	fiona	API to OGR (Vector) layer of GDAL	<a href="http://toblerity.org/fiona/">http://toblerity.org/fiona/</a> (link is external)
Mr. Chaipat @Ayutthaya GIS	rasterio	Reading and writing geospatial raster data	<a href="https://github.com/mapbox/rasterio">https://github.com/mapbox/rasterio</a> (link is external)

67

# Python Geospatial Development Stack

Layer	Package Name	Description	URL
Geoprocessing	shapely	Deterministic spatial analysis	<a href="http://toblerity.org/shapely">http://toblerity.org/shapely</a>
	rasterstats	Summarizing rasters using vector geometries	<a href="https://github.com/perrygeo/python-rasterstats">https://github.com/perrygeo/python-rasterstats</a>
	geopandas	Pandas-like spatial operations on geometric types	<a href="http://geopandas.org">http://geopandas.org</a>
Mr. Chaipat @Ayutthaya GIS	pyproj	PROJ4 Interface for cartographic transformations	<a href="https://github.com/jswhit/pyproj">https://github.com/jswhit/pyproj</a>

68

# Python Geospatial Development Stack

Layer	Package Name	Description	URL
Geo visualization	basemap	Plotting 2D data on maps	<a href="https://github.com/matplotlib/basemap">https://github.com/matplotlib/basemap</a>
	cartopy	Cartographic tools	<a href="http://scitools.org.uk/cartopy">http://scitools.org.uk/cartopy</a>
	folium	Visualization via interactive Leaflet maps	<a href="https://github.com/python-visualization/folium">https://github.com/python-visualization/folium</a>
	Bokeh (geo)	Interactive visualization library browsers	<a href="https://github.com/bokeh/bokeh">https://github.com/bokeh/bokeh</a>
Mr. Chaipat @Ayutthaya GIS	datashader	Big data visualization graphics pipeline	<a href="https://github.com/bokeh/datashader">https://github.com/bokeh/datashader</a>

69

# Python Geospatial Development Stack

Layer	Package Name	Description	URL
Spatial Statistical Analysis	PySAL	Spatial data analysis	<a href="http://pysal.org">http://pysal.org</a>
	pykriging	Geostatistics	<a href="http://pykriging.com/">http://pykriging.com/</a>
	OSMnx + Networkx	network analysis , OpenStreetMap Data Analysis	<a href="https://osmnx.readthedocs.io/en/stable/">https://osmnx.readthedocs.io/en/stable/</a> <a href="https://networkx.github.io/documentation/networkx-1.10/overview.html">https://networkx.github.io/documentation/networkx-1.10/overview.html</a>

Mr. Chaipat @Ayutthaya GIS

70

# Python Geospatial Development Stack

Layer	Package Name	Description	URL
Web & Data Service	OWSlib	Client programming interface to OGC web service	<a href="https://geopython.github.io/OWSLib">https://geopython.github.io/OWSLib</a>
	PyWPS	OGC WPS Framework for geospatial data	<a href="https://pywps.org/">https://pywps.org/</a>
	MapProxy	Open source proxy for geospatial data. , Tile cache & WMS server	<a href="https://mapproxy.org/">https://mapproxy.org/</a>

Mr. Chaipat @Ayutthaya GIS

71

# Python Geospatial Installation

72

## Installation

- ติดตั้งผ่าน pip ได้เกือบทั้งหมด
- บาง lib จะจะต้องมีการ compile source code
- 99% รองรับการทำงานกับ Window OS รองรับ Anapip
- รองรับการทำงานบน Linux / Mac

Mr. Chaipat @Ayutthaya GIS

73

## Installation

### # Install shapely

```
conda install -c conda-forge shapely
```

### # Install geopandas

```
conda install -c conda-forge geopandas
```

### # Install cartopy

```
conda install -c conda-forge cartopy
```

### # Install networkx

```
conda install networkx
```

### # Install osmnx

```
conda install -c conda-forge osmnx
```

### # Install statsmodels

```
conda install statsmodels
```

### # Install folium

```
conda install -c conda-forge folium
```

### # Install bokeh

```
conda install bokeh
```

### # Install shapely

```
pip install shapely
```

### # Install geopandas

```
pip install geopandas
```

### # Install cartopy

```
pip install cartopy
```

### # Install networkx

```
pip install networkx
```

### # Install osmnx

```
pip install osmnx
```

### # Install statsmodels

```
pip install statsmodels
```

### # Install folium

```
pip install folium
```

### # Install bokeh

```
pip install bokeh
```

74

## Installation

- สามารถ import ตัว lib ได้แบบไม่มี error
- หรือ สามารถรัน print() ดูรายละเอียดของการติดตั้งและ package ได้เปลี่ยน ว่าพร้อมจะทำงาน

```
import geopandas
import scipy
import shapely
import matplotlib.pyplot as plt
import pysal
import bokeh
import cartopy
import statsmodels
```

Mr. Chaipat @Ayutthaya GIS

75

# Google Colaboratory

76

## Google Colaboratory

- free Jupyter notebook environment
- Support Python 2 , 3
- Linux OS
- execute code, save and share code & result
- access powerful computing resources {CPU/GPU/TPU}
- Data storage : Google Drive



Mr. Chaipat @Ayutthaya GIS

77

# Google Colaboratory

Mr. Chaipat @Ayutthaya GIS

The screenshot shows the Google Drive interface. A new folder named "Colaboratory" has been created. A context menu is open over this folder, with "More" selected. Under "More", "Colaboratory" is highlighted in the list of available apps.

78

# Google Colaboratory

Welcome to Colaboratory!

Colaboratory is a Google research project created to help disseminate machine learning education and research. It's a Jupyter notebook environment that requires no setup to use and runs entirely in the cloud.

Colaboratory notebooks are stored in Google Drive and can be shared just as you would with Google Docs or Sheets. Colaboratory is free to use.

EXAMPLES RECENT NOTEBOOKS GOOGLE DRIVE GITHUB

Title	First opened	Last opened
Hello, Colaboratory	5 days ago	0 minutes ago
External data: Drive, Sheets, and Cloud Storage	3 days ago	3 days ago
Overview of Colaboratory Features	5 days ago	3 days ago
Example Solution of Final Project.ipynb	4 days ago	3 days ago
Snippets: Importing libraries	4 days ago	4 days ago

NEW PYTHON 3 NOTEBOOK CANCEL

[https://colab.research.google.com/notebooks/basic\\_features\\_overview.ipynb](https://colab.research.google.com/notebooks/basic_features_overview.ipynb)

<https://colab.research.google.com>

79

# Google Colaboratory

Mr. Chaipat @Ayutthaya GIS

The screenshot shows the Google Colaboratory interface. On the left, a sidebar menu is open under the 'File' tab, showing options like 'New Python 2 notebook', 'Open Drive notebook...', and 'Save a copy in Drive...'. The main workspace displays a 'Welcome to Colaboratory!' message and a section titled 'TensorFlow execution'. It includes a mathematical equation and a code snippet demonstrating TensorFlow operations:

```
[1. 1. 1.] + [1. 2. 3.] = [2. 3. 4.  
1. 1. 1.] [4. 5. 6.] [5. 6. 7.]  
  
[ ] import tensorflow as tf  
import numpy as np  
  
with tf.Session():  
    input1 = tf.constant(1.0, shape=[2, 3])  
    input2 = tf.constant(np.reshape(np.arange(1.0, 7.0, dtype=np.float32), (2, 3)))  
    output = tf.add(input1, input2)  
    result = output.eval()  
  
result  
  
array([[ 2.,  3.,  4.],  
       [ 5.,  6.,  7.]], dtype=float32)
```

# Google Colaboratory

Mr. Chaipat @Ayutthaya GIS

The screenshot shows the 'Notebook settings' dialog box. It includes fields for 'Runtime type' (set to 'Python 3'), 'Hardware accelerator' (set to 'GPU'), and a checkbox for 'Omit code cell output when saving this notebook'. At the bottom are 'CANCEL' and 'SAVE' buttons.

The background shows the Google Colaboratory interface with a sidebar titled 'Files' showing a list of files and a code cell containing a script related to Google Drive FUSE:

```
[1] # Load the Drive helper and mount  
2 from google.colab import drive  
3  
4 # This will prompt for authorization.  
5 drive.mount('/content/drive')  
  
D: Mounted at /content/drive  
  
D: 07_structured_data.ipynb      Sample Excel file.xlsx  
BigQuery recipes      script.ipynb  
Colab Notebooks      TFGan tutorial in Colab.txt  
Copy of nima colab.ipynb      to_upload (1).ipynb  
created.txt      to_upload (2).ipynb  
Exported DataFrame sheet.gsheet      to_upload (3).ipynb  
foo.txt      to_upload.ipynb  
Fiddle + Drive FUSE example.ipynb      variables.pickle  
Sample Excel file.gsheet
```

# Google Colaboratory

## Cloud Linux Server

-GPU: 1x Tesla K80 , compute 3.7, having 2496 CUDA cores , 12GB GDDR5 VRAM

-CPU: 1x single core hyper threaded Xeon Processors @2.3Ghz

-RAM: ~12.6 GB Available

-Disk: ~33 GB Available



Ayutthaya GIS Consultant Co., Ltd.

82

# Google Drive

```
from google.colab import drive  
drive.mount('/content/gdrive/')  
data_dir = os.path.join('/content/gdrive', 'My Drive','colab_notebooks')  
os.listdir(data_dir)
```

Mr. Chaipat @Ayutthaya GIS

83

# Spatial Data Model

84

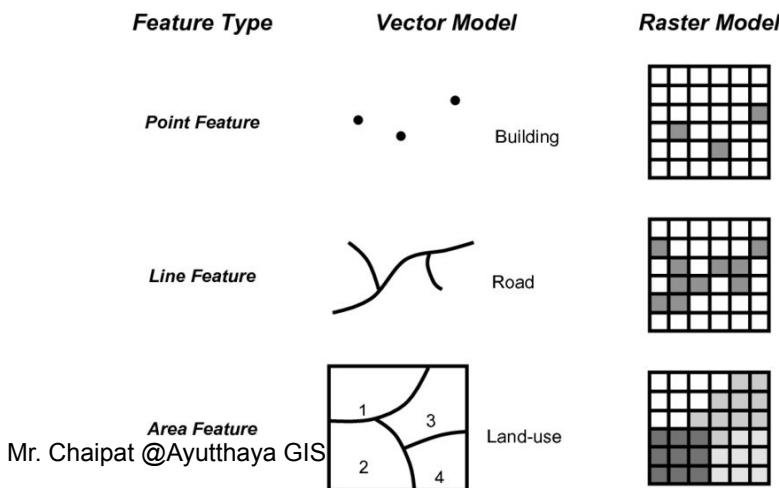
## Spatial Data Model

- การสร้างแบบจำลองข้อมูลเชิงพื้นที่ในระบบสารสนเทศ
- Spatial Data Model
  - a. Vector Data model
  - b. Raster Data Model

Mr. Chaipat @Ayutthaya GIS

85

# Spatial Data Model



86

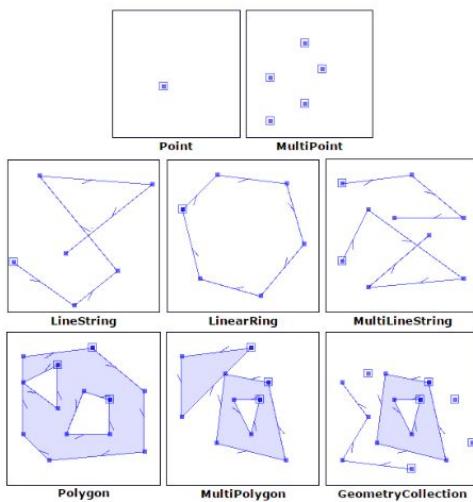
# Spatial Data Model

- Vector Data Model คือ geometric objects รหัสเชิงเลข ที่ใช้นิยามตัวแทนวัตถุในระบบคอมพิวเตอร์
- Geometry Object โดยมีการระบุพิกัดตำแหน่งอ้างอิง และ index เชื่อมโยงกับข้อมูลรายละเอียดที่เกี่ยวข้อง
- สามารถแบ่ง Geometry Object พื้นฐาน ได้แก่ Point , Line และ Polygon
- ยังสามารถแยกย่อยตามระนาบทั้ง 2D และ 3D

Mr. Chaipat @Ayutthaya GIS

87

## Spatial Data Model



88

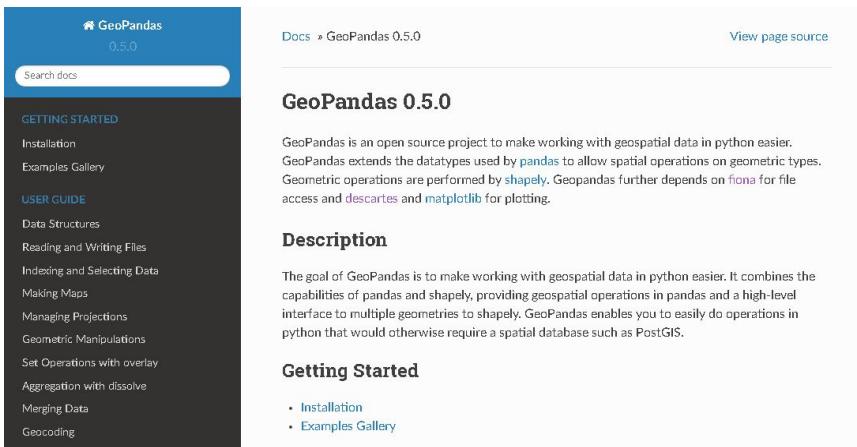
## Spatial Data Model & GeoPandas

- Python มี module ในการจัดการข้อมูล geometry หลายตัว ใน workshop นี้จะเรียนรู้จากการจัดการด้วย **Geopandas**
- GeoPandas เป็น open source package สร้างเพื่อการทำงานกับ ข้อมูล geospatial ลักษณะของการผูกรวม
- โดยการประมวลผลด้าน data model เป็นองหลังมี shapely ( GEOS C++ library) ดำเนินการ
- 

Mr. Chaipat @Ayutthaya GIS

89

# Spatial Data Model & GeoPandas



The screenshot shows the GeoPandas 0.5.0 documentation homepage. The left sidebar has a dark background with white text, listing categories like 'GETTING STARTED' and 'USER GUIDE'. The main content area has a light background with a header 'GeoPandas 0.5.0'. It includes a breadcrumb navigation 'Docs » GeoPandas 0.5.0' and a link 'View page source'. Below the header is a section titled 'GeoPandas 0.5.0' with a brief description of the project's purpose and dependencies. There is also a 'Description' section and a 'Getting Started' section with links to 'Installation' and 'Examples Gallery'.

Mr. Chaipat @Ayutthaya GIS

90

## Point

- “จุด” ตัวแทน ประกอบด้วยตัวเลขพิกัด 2 มิติ หรือ 3 มิติ บน space
- ทดลองทำการสร้าง point จากกลุ่มข้อมูลพิกัด 2 มิติโดยใช้คลาส Geoseries
- จากนั้นทำการสร้าง geometry object สำหรับ Geodataframe บน pandas โดยทำการระบุระบบพิกัดอ้างอิง(CRS) เป็น EPSG:4326

Mr. Chaipat @Ayutthaya GIS

91

## Point

```
#Point
gs = geopandas.GeoSeries([Point(-120, 45), Point(-121.2, 46), Point(-122.9, 47.5)])
gs.crs = {'init': 'epsg:4326'}
print(gs)

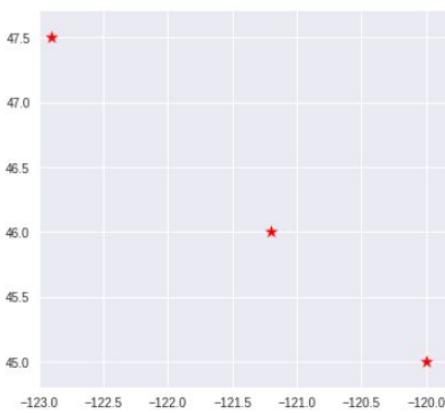
#Plot geoseries
gs.plot(marker='*', color='red', markersize=100, figsize=(6,6))
plt.xlim([-123, -119.8])
plt.ylim([44.8, 47.7]);
```

Mr. Chaipat @Ayutthaya GIS

92

## Point

```
0    POINT (-120 45)
1    POINT (-121.2 46)
2    POINT (-122.9 47.5)
dtype: object
```



Mr. Chaipat @Ayutthaya GIS

93

## Line

- “เล่น” ตัวแทนประกอบด้วย set ของจุดเชื่อมตอกัน
- ทำการสร้าง Linestring object จากกลุ่มข้อมูลของ point แบบ 2 มิติ ซึ่ง point จะเป็นตัวแทนของ node ที่อยู่บน Line ที่สร้างขึ้น

Mr. Chaipat @Ayutthaya GIS

94

## Line

```
# Linestring
mline1 = LineString([
    Point(0, 0),
    Point(0, 1),
    Point(1, 1),
    Point(1, 2),
    Point(3, 4),
    Point(5, 6),
])
mline2 = LineString([
    Point(6, 10),
    Point(10, 14),
    Point(11, 12),
    Point(12, 15),
])

#build line
gs_l = geopandas.GeoSeries([mline1, mline2])
print(gs_l)

#plot geoseries
gs_l.plot()
```

Mr. Chaipat @Ayutthaya GIS

95

## Line

```
0  LINestring (0 0, 0 1, 1 1, 1 2, 3 4, 5 6)
1  LINestring (6 10, 10 14, 11 12, 12 15)
dtype: object
<matplotlib.axes._subplots.AxesSubplot at 0x7f60c52b9358>
```



96

## Polygon

- พื้นที่รูปปิด ตัวแทนของรูปทรงปิด ที่ประกอบด้วย จุดอ้างอิง ตัวแทนตำแหน่ง node และ edge ที่แน่นอน
- ตัวอย่าง ทำการสร้าง Polygon object จากกลุ่มข้อมูลของ Point 2 มิติ ที่เป็นตัวแทนของ node ของ polygon
- จากนั้นใช้ geoseries ในการสร้าง geometry object ประเภท polygon ขึ้นมาโดยจะได้รูปปิด ที่มีทั้ง node และ edge สุดท้ายสามารถนำไปใช้คำนวณคุณสมบัติประกอบอื่นๆ ได้ต่อไป

# Polygon

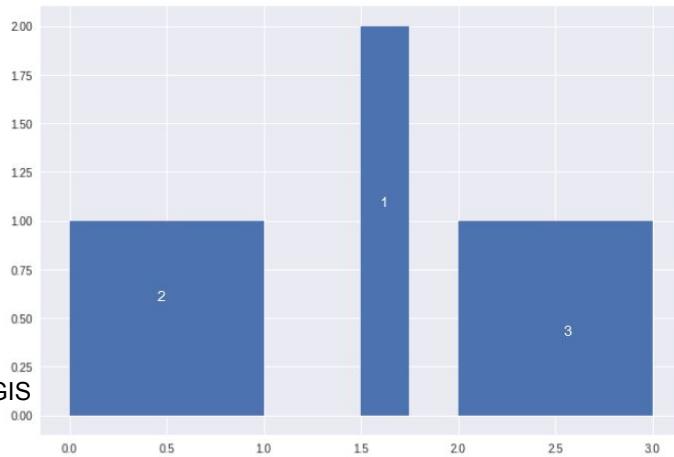
```
#Polygon  
p1 = Polygon([(1.5, 0), (1.75, 0), (1.75, 2),(1.5, 2)])  
p2 = Polygon([(0, 0), (1, 0), (1, 1), (0, 1)])  
p3 = Polygon([(2, 0), (3, 0), (3, 1), (2, 1)])  
gs_po = geopandas.GeoSeries([p1, p2, p3])  
print(gs_po)  
  
#plot geoseries  
gs_po.plot()
```

Mr. Chaipat @Ayutthaya GIS

98

# Polygon

```
0    POLYGON ((1.5 0, 1.75 0, 1.75 2, 1.5 2, 1.5 0))  
1    POLYGON ((0 0, 1 0, 1 1, 0 1, 0 0))  
2    POLYGON ((2 0, 3 0, 3 1, 2 1, 2 0))  
dtype: object  
<matplotlib.axes._subplots.AxesSubplot at 0x7f60c4eae400>
```



Mr. Chaipat @Ayutthaya GIS

99

# Polygon

- พื้นที่รูปปิดจะมีคุณสมบัติ เช่น พื้นที่, เส้นรอบรูป และจุดกึ่งกลาง
- ตัวอย่าง ทำการคำนวณคุณสมบัติเรขาคณิตของ polygon พื้นฐาน  
ได้แก่ Area, BBOX , Centroid

```
#Centroid calculation
cents = gs_po.centroid
print('Centroid \n',cents)

#Area calculation
Area = gs_po.area
print('Area \n',Area)

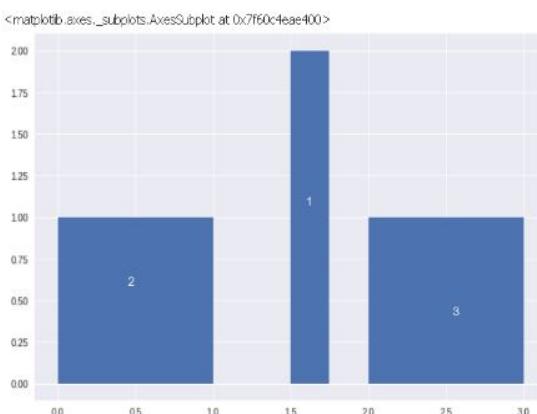
# Bounding box
bbox= gs_po.bounds
print('Bounding box \n',bbox)
```

Mr. Chaipat @Ayutthaya GIS

100

# Polygon

```
Centroid
0 POINT (1.625 1)
1 POINT (0.5 0.5)
2 POINT (2.5 0.5)
dtype: object
Area
0    0.5
1    1.0
2    1.0
dtype: float64
Bounding box
   minx  miny  maxx  maxy
0  1.5  0.0  1.75  2.0
1  0.0  0.0  1.00  1.0
2  2.0  0.0  3.00  1.0
```



Mr. Chaipat @Ayutthaya GIS

101

# Spatial Data Management

102

## Spatial Data Management

- GeoPandas อ่าน/เขียน และจัดการไฟล์ GIS format ผ่าน Fiona (OGR + Proj4)
- Plot หรือแสดงผล geometry ด้วย descartes(เดкар์ต) (matplotlib)
- Pandas จัดการข้อมูล attribute data , table และ sql query
- GeoDataframe

Mr. Chaipat @Ayutthaya GIS

103

## Spatial Data format

- GeoPandas รองรับการทำงานกับ GIS Vector File เกือบทุกประเภท เช่นเดียวกับ OGR
- รองรับประเภทหลัก เช่น Shapefile, GeoJSON, KML และ GPKG

Mr. Chaipat @Ayutthaya GIS

104

## Spatial Data Management

- ทำการเปิดและอ่านไฟล์ province.shp จากโฟเดอร์ data
- ทดสอบผลการอ่านไฟล์ด้วยการแสดงข้อมูลบน table
- แสดงข้อมูล Geometry object แบบ WKT (polygon) พร้อมข้อมูลจาก database(.dbf)
- Plot แผนที่จาก ข้อมูล geometry พร้อมระบบพิกัด

Mr. Chaipat @Ayutthaya GIS

105

# Spatial Data Management

```
In [94]: #load gis shp file
province=gpd.read_file("./data/province_shp")
province.head(3)
```

PROV_CODE	TOT_MALE	TOT_FEMALE	TOT_ALL	VOT_MALE	VOT_FEMALE	VOT_ALL	PROV_NAM_T	PROV_NAM_E	P_CODE	geometry
57	634959	639255	1274214	405515	419521	825036	จังหวัดเชียงราย	Changwat Chiangrai	CHR	POLYGON ((597449.216804 2262033.986135, ...
50	787591	808264	1595855	545188	582667	1127855	จังหวัดเชียงใหม่	Changwat Chiang Mai	CHM	POLYGON ((553594.212661 2227842.917545, ...
58	124295	115719	240014	71744	68597	140341	จังหวัดเชียงใหม่	Changwat Mae Hong Son	MS	POLYGON ((399866.083405 2184810.100011, ...

```
In [100]: province.geometry.bounds
```

	minx	miny	maxx	maxy
0	5.270823e+05	2.100870e+06	6.648943e+05	2.263225e+06
1	3.988433e+05	1.906465e+06	5.598969e+05	2.227888e+06
2	3.252167e+05	1.950139e+06	4.633613e+05	2.191310e+06
3	5.716720e+05	2.079831e+06	6.713170e+05	2.182879e+06
4	6.408247e+05	1.992243e+06	7.480584e+05	2.172110e+06
5	4.872655e+05	1.902434e+06	6.186460e+05	2.147156e+06
6	5.389370e+05	1.955443e+06	6.638243e+05	2.082903e+06
7	4.652020e+05	1.926472e+06	5.338147e+05	2.068751e+06
8	8.234012e+05	1.948806e+06	1.050002e+06	2.045524e+06
9	5.948195e+05	1.895027e+06	7.318766e+05	2.033675e+06

```
: province.plot(figsize=(6,5));
```



106

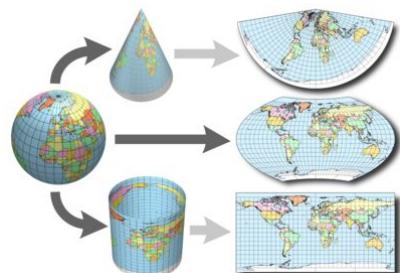
# Spatial Reference System

107

## Spatial Reference System(SRS)

- Aka coordinate reference system (CRS)
- การนิยามระบบพิกัดอ้างอิงของ Geometry model ในระบบที่ใช้เป็นตัวแทนวัตถุบนพื้นโลก
- Map Projection = โมเดลการฉายผิวโลกทรงโค้ง 3D ให้กลายเป็นแผนที่(Map) 2 มิติ

Mr. Chaipat @Ayutthaya GIS



108

## Spatial Reference System(SRS)

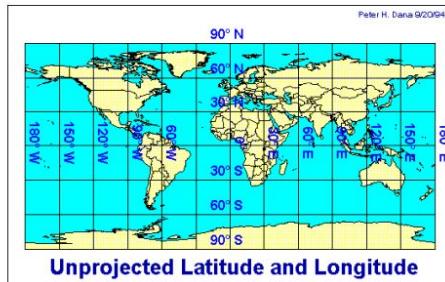
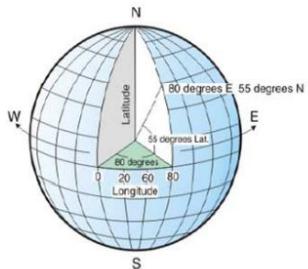
- ระบบพิกัดอ้างอิง (coordinate system) ของข้อมูลเชิงพื้นที่
- การระบุพิกัดตำแหน่งอ้างอิงบนแผนที่
  - >Geographic coordinate system (latitude, longitude)
  - >Universal Transverse Mercator (UTM)

Mr. Chaipat @Ayutthaya GIS

109

# Geographic coordinate system

- การระบุพิกัดตำแหน่งของจุดบนแผนที่แบบ latitude, longitude



Mr. Chaipat @Ayutthaya GIS

110

# Geographic coordinate system

## EPSG:4326

WGS 84 -- WGS84 - World Geodetic System 1984, used in GPS

### Attributes

Unit: degree (supplier to define representation)

Geodetic CRS: WGS 84

Datum: World Geodetic System 1984

Ellipsoid: WGS 84

Prime meridian: Greenwich

Data source: OGP

Information source: EPSG. See 3D CRS for original information source.

Revision date: 2007-08-27

```
GEOGCS["WGS 84",
  DATUM["WGS_1984",
    SPHEROID["WGS 84",6378137,298.257223563,
      AUTHORITY["EPSG","7030"]],
    AUTHORITY["EPSG","6326"]],
  PRIMEM["Greenwich",0,
    AUTHORITY["EPSG","8901"]],
  UNIT["degree",0.0174532925199433,
    AUTHORITY["EPSG","9122"]],
  AUTHORITY["EPSG","4326"]]
```

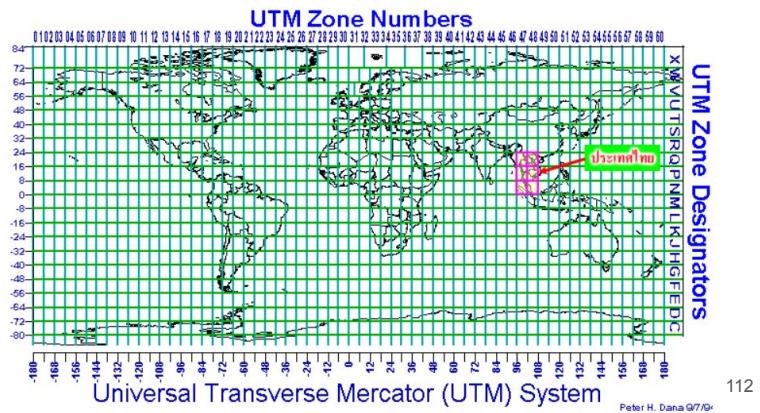
Mr. Chaipat @Ayutthaya GIS

111

## Universal Transverse Mercator (UTM)

- การระบุพิกัดตำแหน่งอ้างอิงบนแผนที่แบบ GRID
- เมืองไทยมีพิกัดตอกบนกริด UTM โซน 47 และ 48 มีรหัสเลขช่วง N, P และ Q
- ตัวอย่าง 47P356482

Mr. Chaipat @Ayutthaya GIS



112

Peter H. Dana ©7/04

## พื้นหลักฐานอ้างอิง (Datum)

- ระบบหลักฐานที่ใช้อ้างอิงในการกำหนดค่าทางแนวตั้งและแนวราบ
- เพื่อใช้เป็นจุดบันทึกทางความสูงและความคุณตำแหน่งบนแผนที่ สำหรับอ้างอิงในการหาตำแหน่งบนพื้นผิวโลก
- Horizontal Datum
  - > Local Datum : ไทยใช้ Indian Datum 1975
  - > Global Datum WGS 84 (World Geodetic System 1984) แห่ง
- Vertical Datum : ไทยใช้ ค่าอ้างอิงจากระดับน้ำทะเล平原กลาง ณ เกาะหลัก จังหวัดปราจวบ

Mr. Chaipat @Ayutthaya GIS

113

# Universal Transverse Mercator (UTM)

- การระบุพิกัดตำแหน่งอ้างอิงบนแผนที่แบบ GRID
- เมืองไทยมีพิกัด SRS ประเภท UTM ที่พบได้แก่
- EPSG 32647 ,
- EPSG 24047
- EPSG 32648,
- EPSG 24048

Mr. Chaipat @Ayutthaya GIS

114

## EPSG:32647 WGS 84 / UTM zone 47N

### EPSG:32647

WGS 84 / UTM zone 47N

#### Attributes

Unit: metre

Geodetic CRS: WGS 84

Datum: World Geodetic System 1984

Ellipsoid: WGS 84

Prime meridian: Greenwich

Data source: OGP

Revision date: 2006-08-25

```
PROJCS["WGS 84 / UTM zone 47N",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
            SPHEROID["WGS 84",6378137.298257223563,
                AUTHORITY["EPSG","7030"]],
            AUTHORITY["EPSG","6326"]],
        PRIMEM["Greenwich",0,
            AUTHORITY["EPSG","8901"]],
        AUTHORITY["EPSG","9901"]],
    UNIT["degree",0.0174532925199433,
        AUTHORITY["EPSG","9122"]],
    AUTHORITY["EPSG","4326"]],
    PROJECTION["Transverse_Mercator"],
    PARAMETER["latitude_of_origin",0],
    PARAMETER["central_meridian",99,
        AUTHORITY["EPSG","9001"]],
    PARAMETER["scale_factor",0.9996,
        AUTHORITY["EPSG","9002"]],
    PARAMETER["false_easting",500000,
        AUTHORITY["EPSG","9003"]],
    PARAMETER["false_northing",0,
        AUTHORITY["EPSG","9004"]],
    UNIT["metre",1,
        AUTHORITY["EPSG","9001"]],
    AXIS["Easting",EAST,
        AUTHORITY["EPSG","9101"]],
    AXIS["Northing",NORTH,
        AUTHORITY["EPSG","9102"]]]
```

115

# EPSG:24047 Indian 1975 / UTM zone 47N

## Attributes

Unit: metre  
Geodetic CRS: Indian 1975  
Datum: Indian 1975  
Ellipsoid: Everest 1830 (1937 Adjustment)  
Prime meridian: Greenwich  
Data source: OGP  
Revision date: 1995-06-02

```
PROJCS["Indian 1975 / UTM zone 47N",
GEOGCS["Indian 1975",
DATUM["Indian_1975",
SPHEROID["Everest 1830 (1937 Adjustment)",6377276.345,300.8017,
AUTHORITY["EPSG","7015"]],
TO_WGS84[210.814289,0.0,0.0],
AUTHORITY["EPSG","6240"]],
PRIMEM["Greenwich",0,
AUTHORITY["EPSG","8901"]],
UNIT["degree",0.0174532925199433,
AUTHORITY["EPSG","9122"]],
AUTHORITY["EPSG","4240"]],
PROJECTION["Transverse_Mercator"],
PARAMETER["latitude_of_origin",0],
PARAMETER["central_meridian",99],
PARAMETER["scale_factor",0.9996],
PARAMETER["false_easting",500000],
PARAMETER["false_northing",0],
UNIT["metre",1,
AUTHORITY["EPSG","9001"]],
AXIS["Easting",EAST],
AXIS["Northing",NORTH],
AUTHORITY["EPSG","24047"]]
```

116

# EPSG:32648 WGS 84 / UTM zone 48N

## EPSG:32648

### WGS 84 / UTM zone 48N

## Attributes

Unit: metre  
Geodetic CRS: WGS 84  
Datum: World Geodetic System 1984  
Ellipsoid: WGS 84  
Prime meridian: Greenwich  
Data source: OGP  
Revision date: 2006-08-25

```
PROJCS["WGS 84 / UTM zone 48N",
GEOGCS["WGS 84",
DATUM["WGS_1984",
SPHEROID["WGS 84",6378137,298.257223563,
AUTHORITY["EPSG","7030"]],
AUTHORITY["EPSG","6326"]],
PRIMEM["Greenwich",0,
AUTHORITY["EPSG","8901"]],
UNIT["degree",0.0174532925199433,
AUTHORITY["EPSG","9122"]],
AUTHORITY["EPSG","4326"]],
PROJECTION["Transverse_Mercator"],
PARAMETER["latitude_of_origin",0],
PARAMETER["central_meridian",105],
PARAMETER["scale_factor",0.9996],
PARAMETER["false_easting",500000],
PARAMETER["false_northing",0],
UNIT["metre",1,
AUTHORITY["EPSG","9001"]],
AXIS["Easting",EAST],
AXIS["Northing",NORTH],
AUTHORITY["EPSG","32648"]]
```

117

# EPSG:900913 Google Maps Global Mercator

## EPSG:900913

Google Maps Global Mercator -- Spherical Mercator  
(unofficial - used in open source projects / OSGEO)

The unofficial code for Spherical Mercator (Google Maps, OpenStreetMap, ...  
OpenLayers and other open-source projects).

See the official coordinate system [EPSG:3857](#)

Alternatives codes : 3857

Attributes

Data source: PROJ.4

Information source: other

Description: +proj=merc +a=6378137

+lat\_ts=0.0 +lon\_0=0.0 +x\_0=0.0 +y\_0=0  
+nadgrids=@null +wktext +no\_defs

```
PROJCS["Google Maps Global Mercator",
    GEOGCS["WGS 84",
        DATUM["WGS_1984",
            SPHEROID["WGS 84",6378137.298257223563,
                AUTHORITY["EPSG","7030"]],
            AUTHORITY["EPSG","6326"]],
        PRIMEM["Greenwich",0,
            AUTHORITY["EPSG","8901"]],
        UNIT["degree",0.01745329251994328,
            AUTHORITY["EPSG","9122"]],
        AUTHORITY["EPSG","4326"]],
    PROJECTION["Mercator_2SP"],
    PARAMETER["standard_parallel_1",0],
    PARAMETER["latitude_of_origin",0],
    PARAMETER["central_meridian",0],
    PARAMETER["false_easting",0],
    PARAMETER["false_northing",0],
    UNIT["Meter",1],
    EXTENSION["PROJ4","+proj=merc +a=6378137 +b=6378137 +lat_ts=0.0 +lon_0=0.0 +x_0=0.0 +y_0=0 +k=1.0 +units=m +nadgrids=@null
AUTHORITY["EPSG","900913"]]
```

118

# Geocode

119

## Geocode

- การแปลงจากชื่อ ที่อยู่ (address) ให้กลายเป็น ระบบพิกัดบนโลก
- รวมถึงข้อมูล IP address
- โดยมีบริการ(service) ในการแปลง ทำหน้าที่ค้นหาที่อยู่จากฐานข้อมูล spatial database และคำนวนพิกัดตำแหน่ง ส่งกลับมายังปลายทาง
- เช่น Google Geocoding API , OSM Map API, Bing เป็นต้น
- Python modules ได้แก่ Geopy, Geocoder , Geopandas

Mr. Chaipat @Ayutthaya GIS

120

## OGC Web Feature Service

121

## WFS

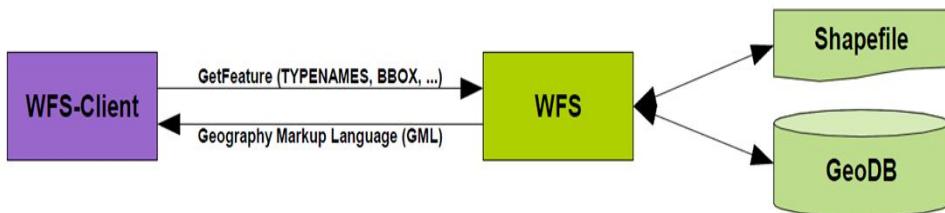
- Open Geospatial Consortium (OGC)
- Web feature service โปรโตคอลมาตรฐานของ OGC พัฒนาขึ้นเพื่อ บริการข้อมูล spatial data
- เชื่อมโยงการทำงานระหว่างระบบ GIS แบบ interoperability ในการ เข้าถึงข้อมูลผ่าน web service
- ข้าม platform , ข้าม software , ข้าม data format
- อ้างอิงเพิ่มเติม

<https://www.opengeospatial.org/standards/wfs>

Mr. Chaipat @Ayutthaya GIS

122

## WFS

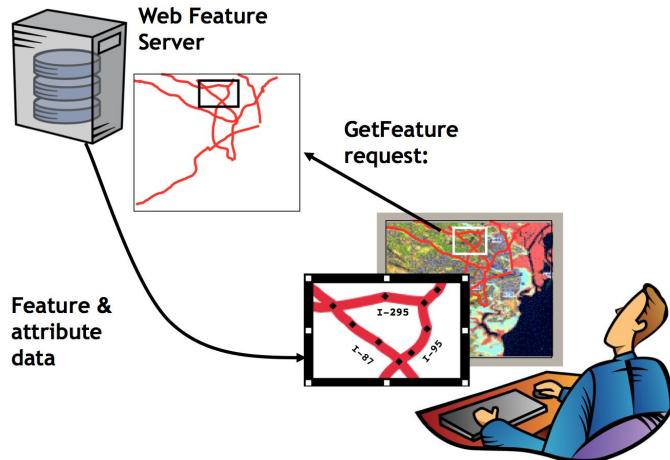


Mr. Chaipat @Ayutthaya GIS

123

# WFS

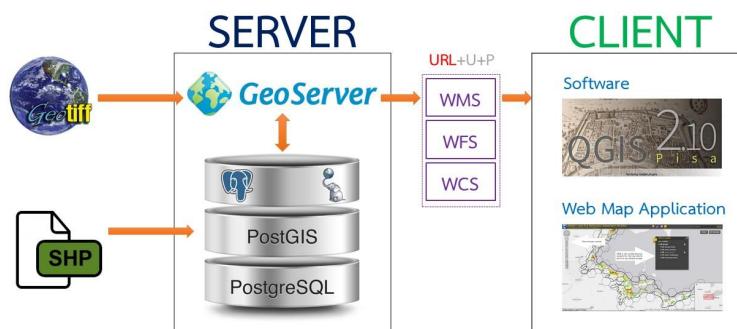
Web Feature Service (WFS) returns data.



Mr. Chaipat @Ayutthaya GIS

124

# WFS



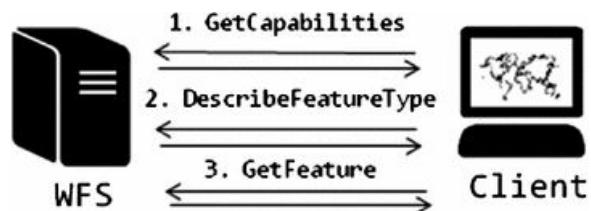
Mr. Chaipat @Ayutthaya GIS

125

## WFS

- โปรโตคอลการทำงานเบื้องต้นของ OGC WFS
- ประกอบด้วย 3 Operation หลักคือ

- 1.GetCapabilities
- 2.DescribeFeatureType
- 3.GetFeature



Mr. Chaipat @Ayutthaya GIS

126

## WFS - GetCapabilities

- Operation เพื่อส่งคำร้องสืบค้าไปยัง WFS server เพื่อตรวจสอบสถานะและข้อมูลที่ให้บริการ
- *<http://localhost/geoserver/ows?service=WFS&version=1.0.0&request=GetCapabilities>*

Mr. Chaipat @Ayutthaya GIS

127

# WFS - GetCapabilities

```
<WFS_Capabilities version="1.0.0" xsi:schemaLocation="http://www.opengis.net/wfs
http://localhost:8080/geoserver/schemas/wfs/1.0.0/WFS-capabilities.xsd">
  <Service>
    <Name>My GeoServer WFS</Name>
    <Title>GeoSolutions GeoServer workshop WFS</Title>
  <Abstract>
    This is the reference implementation of WFS 1.0.0 and WFS 1.1.0, supports all WFS operations
    including transactions.
  </Abstract>
  <Keywords>WFS, WMS, GEOSERVER</Keywords>
  <OnlineResource>http://localhost:8080/geoserver/wfs</OnlineResource>
  <Fees>NONE</Fees>
  <AccessConstraints>NONE</AccessConstraints>
</Service>
<Capability>
  <Request>
    <GetCapabilities>
      <DCPType>
        <HTTP>
          <Get onlineResource="http://localhost:8080/geoserver/wfs?request=GetCapabilities"/>
        </HTTP>
      </DCPType>
    <DCPType>
      <HTTP>
        <Post onlineResource="http://localhost:8080/geoserver/wfs"/>
      </HTTP>
    </DCPType>
  </GetCapabilities>
  <DescribeFeatureType>
    <SchemaDescriptionLanguage>
      <XMLSCHEMA/>
    <SchemaDescriptionLanguage>
      <DCPType>
        <HTTP>
          <Get onlineResource="http://localhost:8080/geoserver/wfs?request=DescribeFeatureType"/>
        </HTTP>
      </DCPType>
    </SchemaDescriptionLanguage>
  </DescribeFeatureType>
</Capability>
```

128

# WFS - DescribeFeatureType

- Operation เพื่อส่งคำร้องสืบค้าไปยัง WFS server เพื่อเข้าถึงข้อมูลรายละเอียดและคำอธิบายของข้อมูลแต่ละ layer
- [http://localhost/geoserver/ows?service=WFS  
&version=1.0.0  
&request=DescribeFeatureType  
&typename=geosolutions:Counties](http://localhost/geoserver/ows?service=WFS&version=1.0.0&request=DescribeFeatureType&typename=geosolutions:Counties)

Mr. Chaipat @Ayutthaya GIS

129

## WFS - DescribeFeatureType

```
- <xsd:schema elementFormDefault="qualified" targetNamespace="http://www.geo-solutions.it
  /workshop">
  <xsd:import namespace="http://www.opengis.net/gml" schemaLocation="http://localhost:8080
  /geoserver/schemas/gml/2.1.2/feature.xsd"/>
- <xsd:complexType name="CountiesType">
- <xsd:complexContent>
- <xsd:extension base="gml:AbstractFeatureType">
- <xsd:sequence>
  <xsd:element maxOccurs="1" minOccurs="0" name="the_geom" nillable="true"
  type="gml:MultiPolygonPropertyType"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="STATEFP10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="COUNTYFP10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="COUNTYNS10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="GEOID10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="NAME10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="NAMELSAD10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="LSAD10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="CLASSFP10" nillable="true"
  type="xsd:string"/>
  <xsd:element maxOccurs="1" minOccurs="0" name="MTFCC10" nillable="true"
  type="xsd:string"/>
```

130

## WFS - GetFeature

- Operation เพื่อส่งคำร้องสืบค้าไปยัง WFS server เพื่อเข้าถึงข้อมูล Spatial data ที่ต้องการ โดยการสืบค้น ต้องระบุพารามิเตอร์ สำคัญ เช่น ชื่อชั้นข้อมูล และสามารถกำหนด parameter เพิ่มเติมในการกรองเช่น bbox,
- ผลลัพธ์ ปกติจะเป็น GML3 แต่มีทางเลือกอื่นๆ เช่น กับ server เช่น geojson , wkb เป็นต้น

Mr. Chaipat @Ayutthaya GIS

131

# WFS - GetFeature

- Operation
  - [http://localhost/geoserver/ows?service=WFS  
&version=1.0.0  
&request=GetFeature  
&typeName=geosolutions:Counties  
&maxFeatures=50  
&outputFormat=GML3](http://localhost/geoserver/ows?service=WFS&version=1.0.0&request=GetFeature&typeName=geosolutions:Counties&maxFeatures=50&outputFormat=GML3)

Mr. Chaipat @Ayutthaya GIS

132

# WFS - GetFeature

Mr. Chaipat @Ayutthaya GIS

133

## **WFS - Server**

WFS in thailand (gov)

<http://fgds-backend.ngis.go.th/geoserver/geonode/ows?>

<http://qservices.gistda.or.th/geoserver/wfs>

<http://mapservices.energy.go.th:80/geoserver/ows?>

134

# Spatial Data Analysis

135

# Spatial Data Analysis

- การวิเคราะห์ข้อมูลเชิงพื้นที่ ประกอบด้วย

## 1. Geoprocessing

คือ การประมวลผล เพื่อสร้างข้อมูล geometry ใหม่จากข้อมูล geometry ตั้งต้น และส่วนการประมวลผล ระหว่าง geometry เพื่อสร้างผลลัพธ์ใหม่

> function การแปลง geometry data และสร้างผลลัพธ์ข้อมูล attribute ใหม่

> Buffer , Merge , Difference, Intersection, Union เป็นต้น

> ศึกษาเพิ่มเติมจาก <http://geopandas.org/reference.html>

## • 2. Spatial Relation

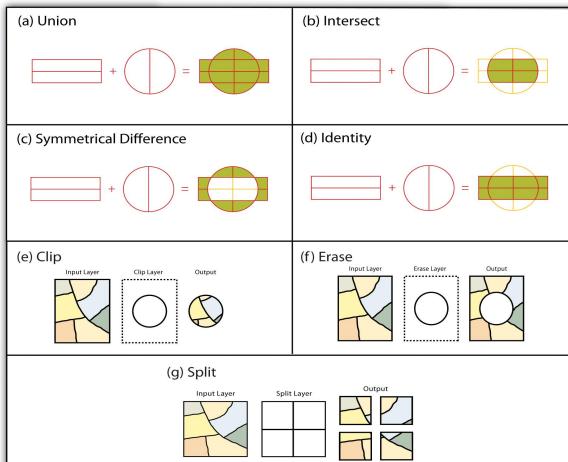
คือความสัมพันธ์ระหว่าง geometry ตัวบ่งบอกคุณลักษณะ และคุณสมบัติข้อมูล สำหรับการวิเคราะห์

136

# Spatial Data Analysis

- การวิเคราะห์ข้อมูลเชิงพื้นที่ ยังรวมไปถึงการวิเคราะห์เชิงลึกอีกด้วย
- เช่น Spatial temporal analysis, Terrain analysis, Surface Analysis: Spatial Interpolation หรือ spatial statistic analysis เป็นต้น

# Geoprocessing



[saylordotorg.github.io](https://saylordotorg.github.io)

138

# Spatial Query

139

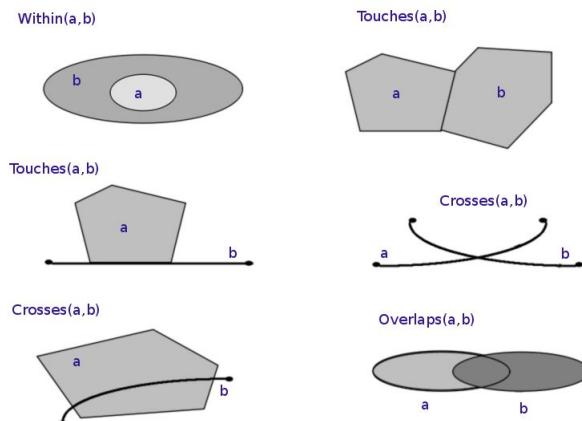
# Spatial Query

- กระบวนการลีบคันข้อมูลเชิงตำแหน่ง ที่ใช้เงื่อนไขเชิงตำแหน่งในการวิเคราะห์ข้อมูลเพื่อให้ได้ผลลัพธ์ตามต้องการ
- นำ geometry data มาร่วมในการสร้างเงื่อนไขเพื่อทำการวิเคราะห์ หรือลีบคันหาข้อมูล จากฐานข้อมูล
- ค้นหาด้วยเงื่อนไข จาก spatial relation function ได้แก่
- contains(), crosses(), .intersects(), overlaps(), touches(), disjoint(), within()

Mr. Chaipat @Ayutthaya GIS

140

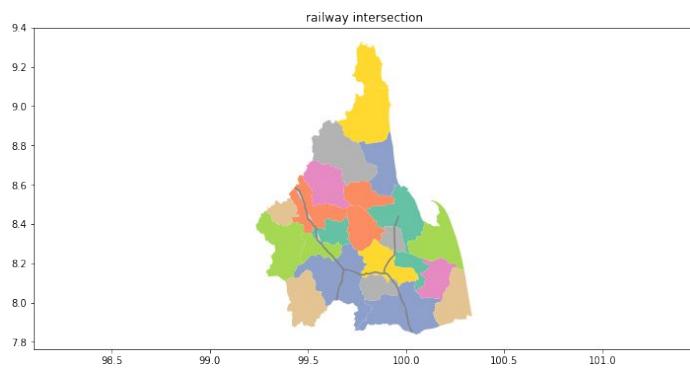
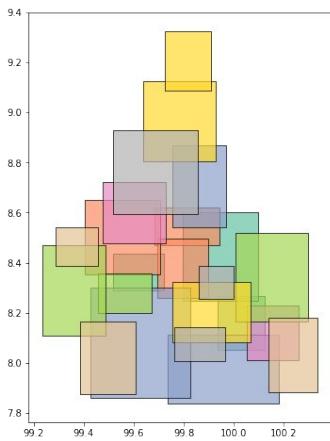
# Spatial relation



Mr. Chaipat @Ayutthaya GIS

141

## Spatial Query



Mr. Chaipat @Ayutthaya GIS

142

## Thematic map

143

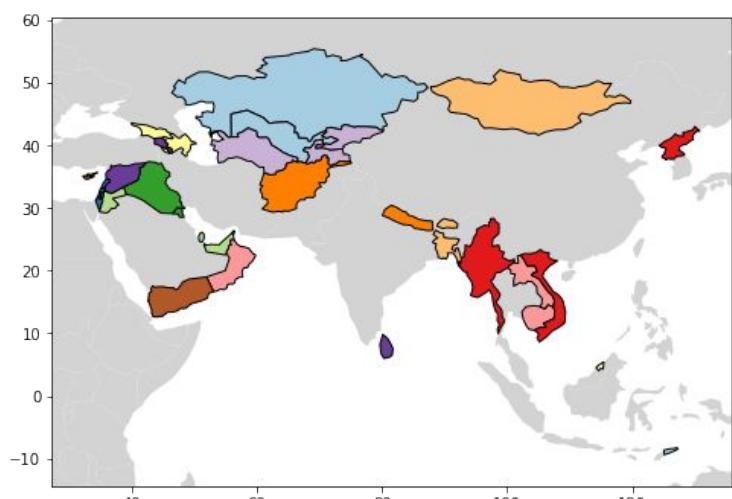
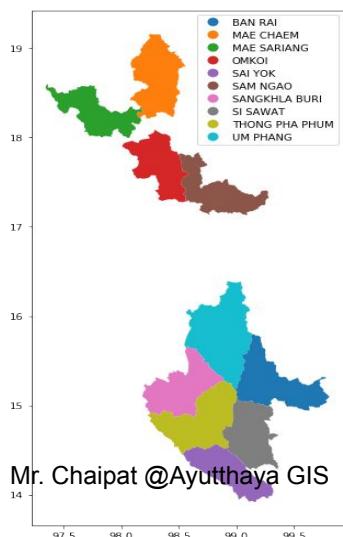
## Thematic map

- การสร้างแผนที่เฉพาะเรื่อง จากการกำหนดเงื่อนไข ในฐานข้อมูล GIS
- นำผลลัพธ์จากการวิเคราะห์ข้อมูล มาสร้างเป็นแผนที่เฉพาะทาง
- อธิบายเล่าเรื่อง โดยกำหนด ลักษณะการแสดงผล สี สัญลักษณ์ ตามค่า(value)
- จากเงื่อนไขบน logical operation และ spatial operation

Mr. Chaipat @Ayutthaya GIS

144

## Thematic map



145

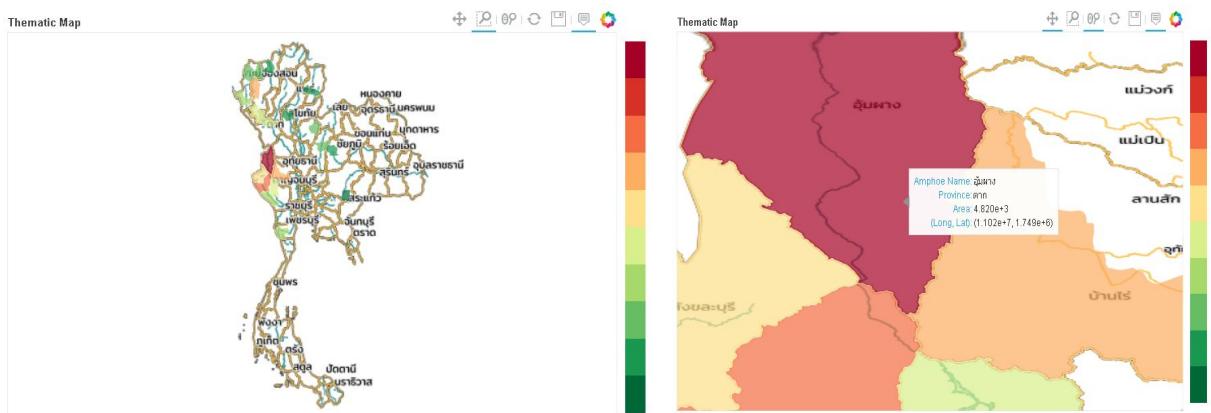
# Web Mapping Application

- การพัฒนาระบบบริการข้อมูล GIS แบบออนไลน์ ผ่านอินเทอร์เน็ต
- รองรับการแสดงผล และการสืบค้นข้อมูล
- Python มีหลายโมดูลที่สามารถเลือกใช้งานได้
- ตัวอย่างใช้ Geopandas + Bokeh
- นอกจากนี้ มีการเชื่อมโยงและพัฒนาการแสดงผลข้อมูลกับ Map service ทั้ง WMS, WMST

Mr. Chaipat @Ayutthaya GIS

146

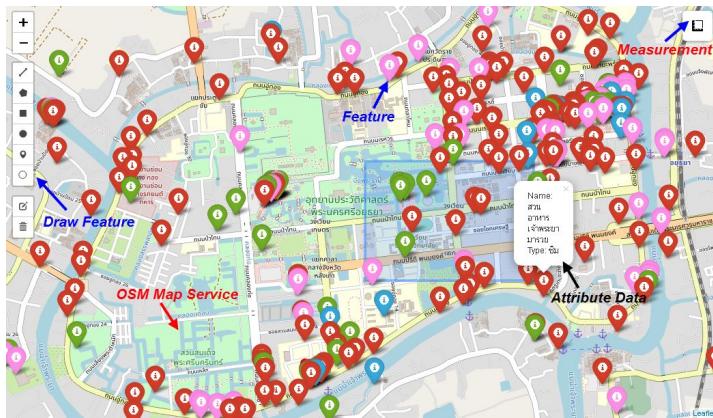
# Web Mapping Application



Mr. Chaipat @Ayutthaya GIS

147

## Web Mapping Application



Mr. Chaipat @Ayutthaya GIS

148

## Open Street Map

149

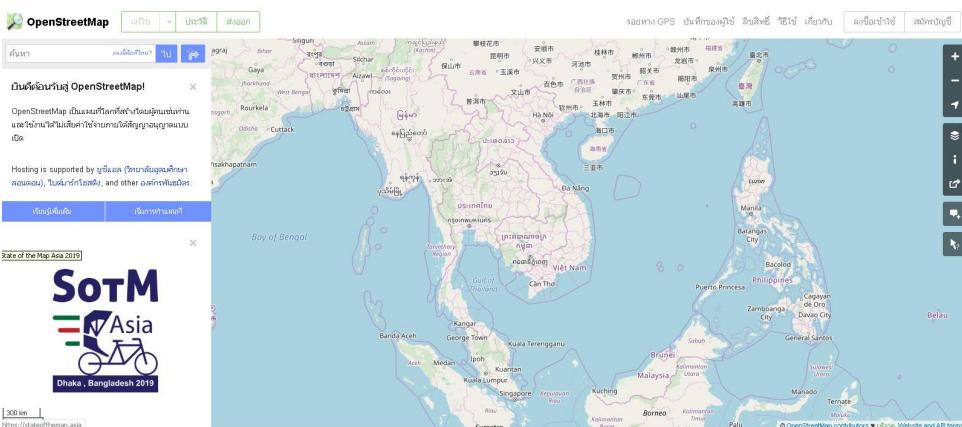
## OSM

- โครงการความร่วมมือเพื่อสร้างแผนที่แบบเปิด
- Crowd-sourced dataset แนวคิดการทำงานร่วมกัน ลักษณะชวนคนเข้ามาร่วมสร้างข้อมูล GIS และพัฒนาฐานข้อมูลเชิงพื้นที่แบบเปิด แต่สามารถตรวจสอบคุณภาพได้
- เป้าหมายเพื่อให้เกิดการใช้งานร่วมกัน การร่วมมือลงแรงปรับปรุงสร้างแผนที่ฐาน เพื่อการพัฒนา
- ปัจจุบันผ่านมากกว่า 10 ปี OSM ยิ่งเติบโต และมีข้อมูลเพิ่มขึ้นมหาศาล ทั่วโลก จำนวนผู้ใช้กว่า 4 million users
- 

Mr. Chaipat @Ayutthaya GIS

150

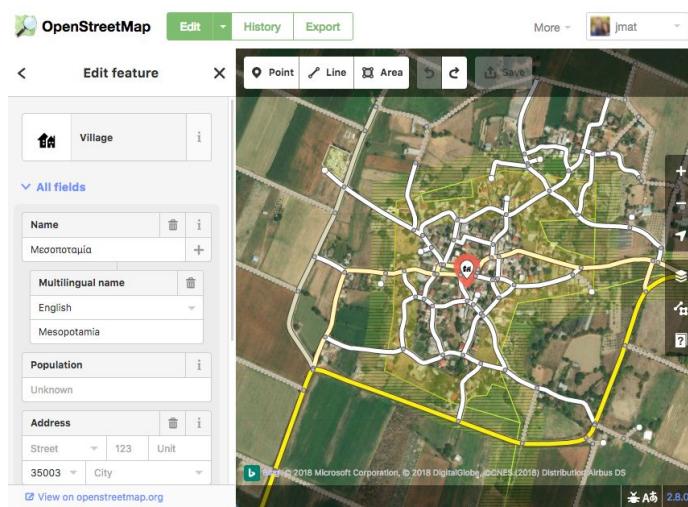
## OSM Web Application



Mr. Chaipat @Ayutthaya GIS

151

## สร้าง/นำเข้าข้อมูล



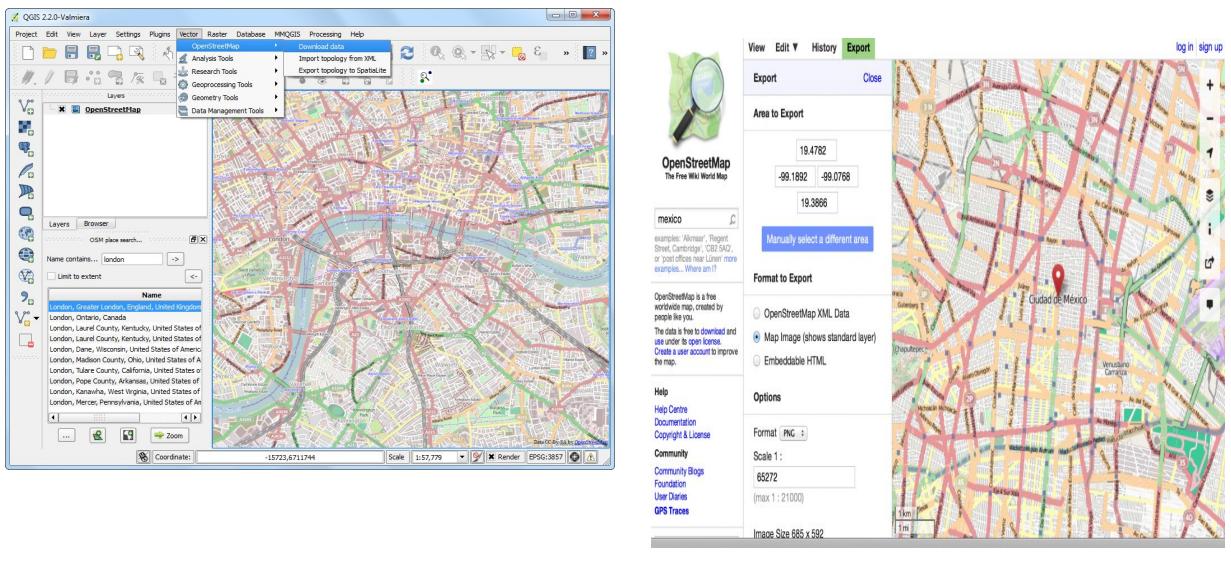
152

## นำเข้าข้อมูลจาก GPS Device

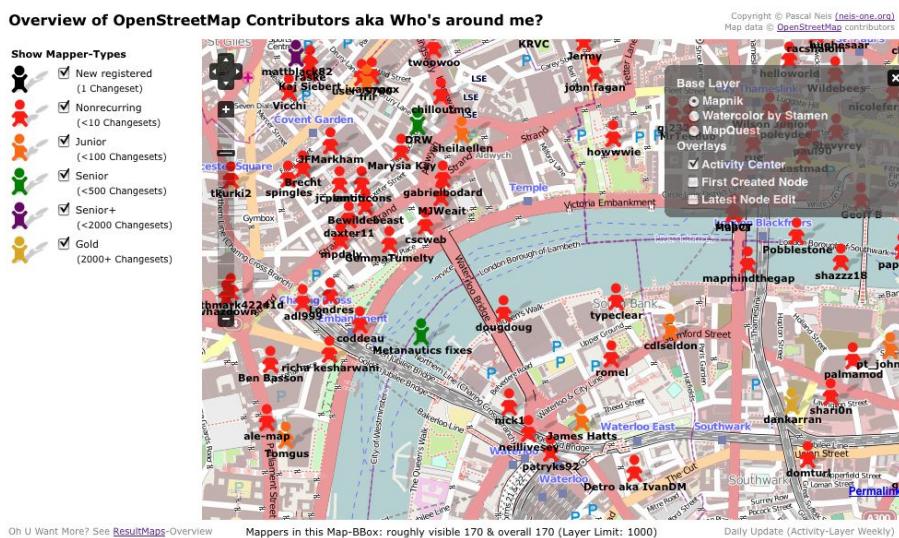


153

# ดาวน์โหลดข้อมูล



# อาสาสมัครทั่วโลก



## การต่อยอดนำไปใช้



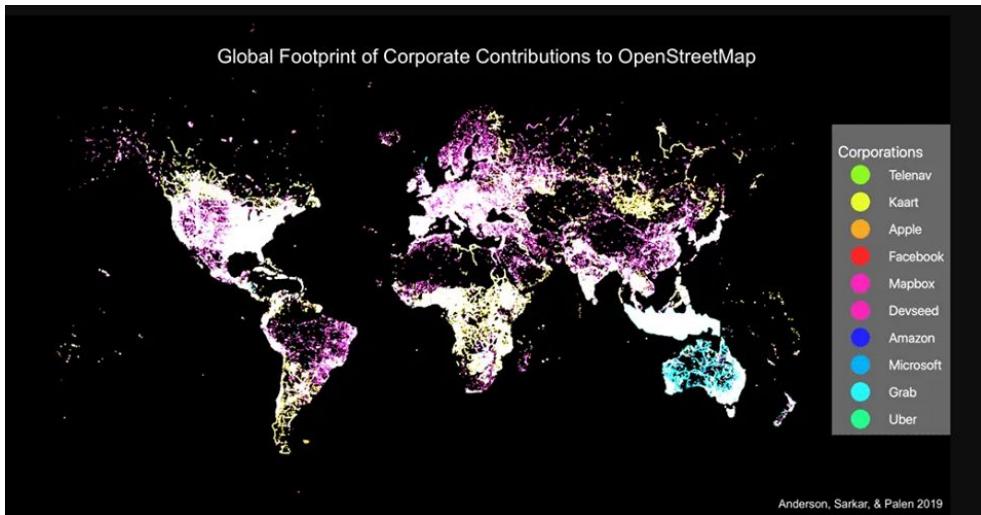
157

## การต่อยอดนำไปใช้



158

## ส่วนร่วมของภาคธุรกิจ



163

## ส่วนร่วมของภาคธุรกิจ

### Grab Data Projects



#### Information about:

- About Grab
- Data Improvement Projects
- Road Mapping Process and Guidelines
- Grab Data Team
- Sources Referred

#### Additional Information

For more information, suggestions or questions contact our Outreach and Community Manager, Jinal Foflia. You can also refer our [OpenStreetMap wiki page](#) for information.

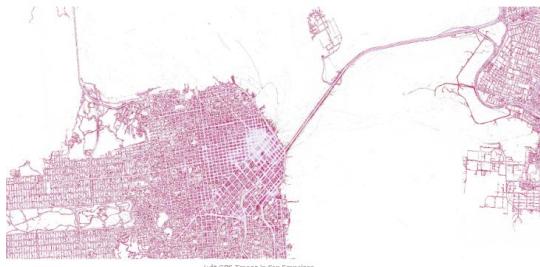
164

## ส่วนร่วมของภาคธุรกิจ

How Lyft Creates Hyper-Accurate Maps from Open-Source Maps and Real-Time Data

 Albert Yuen [Follow](#)  
Sep 7 · 10 min read

By Albert Yuen, James Murphy, Sumanth Ravipati, Deeksha Goyal, Han Kim, Adithya Hemakumar, Milo Han, Alex Ung, Clare Corbett



166

## Facebook Map with AI

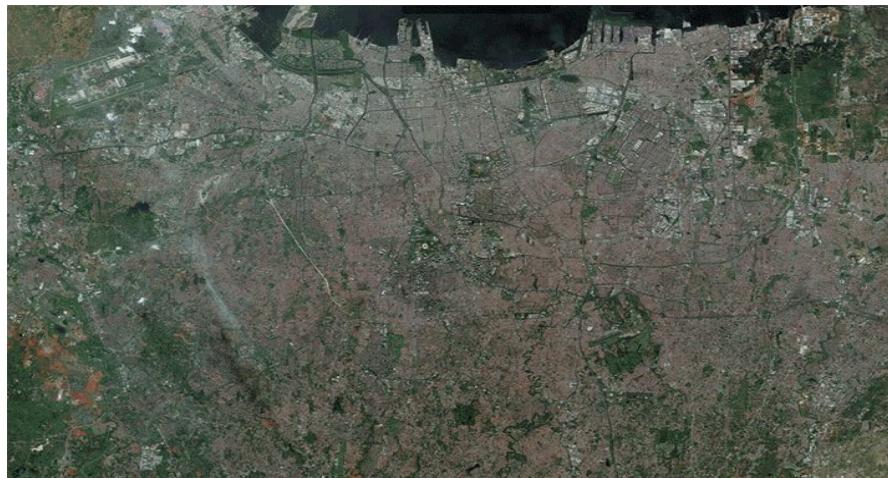
### Using AI to Map the World

In conjunction with OpenStreetMap, we are creating tools that we hope will enable the mapping community to enjoy a faster & more accurate mapping experience. To this end, we used artificial intelligence to predict features on high-resolution satellite imagery; these features are then populated in our RapiD map editing tool. When combined with global-scale humanitarian efforts, we hope RapiD can help realize our vision of a more connected world.



167

## **Facebook Map with AI**



168

**Question ???**

170