

地理空間情報とは



GISと地理空間情報

GISとは

“地理情報システム（GIS : Geographic Information System）は、地理的位置を手がかりに、**位置に関する情報を持ったデータ（空間データ）**を総合的に管理・加工し、視覚的に表示し、高度な分析や迅速な判断を可能にする技術である。”

地理空間情報とは

“地理空間情報とは、空間上の特定の地点又は区域の位置を示す情報（**位置情報**）とそれに**関連付けられた様々な事象に関する情報**、もしくは位置情報のみからなる情報をいう。地理空間情報には、地域における自然、災害、社会経済活動など特定のテーマについての状況を表現する土地利用図、地質図、ハザードマップ等の主題図、都市計画図、地形図、地名情報、台帳情報、統計情報、空中写真、衛星画像等の多様な情報がある。地理空間情報は、その位置情報をキーにして異なるデータを重ね合わせることで、分析等の活用がなされることから、様々な主体によって整備されるデータ間で位置情報の統合がとれている必要がある。”

デジタルツイン ≠ メタバース

Several white lines of varying lengths and thicknesses are drawn diagonally across the bottom right corner of the slide, pointing towards the top right.

デジタルツインは原寸(縮尺1/1)の世界

Several white lines of varying lengths and angles are positioned in the bottom right corner of the slide, creating a dynamic, abstract graphic element.

デジタルツイン = シミュレーションの世界

現実空間をコピーした仮想空間を再現すること

シミュレーションやフィードバックを可能にする仕組み

メタバース = コンピューターの世界

必ずしも既存の現実空間を再現する必要はない

コンピュータの中に構築された3次元の仮想空間

コミュニケーションが行えるサービス・プロダクト

3D都市空間はデジタルツイン



位置情報データの3要素

地物は、ポイント（点）、ライン（線）、ポリゴン（面）で表現される

↓ POINT(点)

← 場所、施設、山頂点、気象観測点

↓ LINESTRING(線)

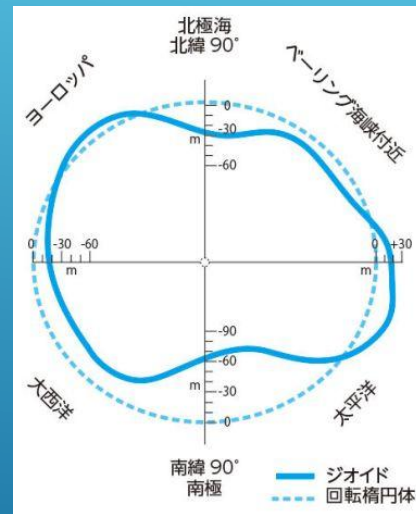
← 河川、道路、鉄道、電話線、河川、上下水管

↓ POLYGON(面)

← 建物、地域、湖、運動場

地球は丸くない？

地表は平らではなく、山や谷があるので表面は凸凹である



ジオイドとは何か？ [前編]

https://qzss.go.jp/overview/column/geoid_151225.html

測地系と座標系

測地系とは

位置を経度、緯度及び標高を表す時の前提条件

座標系とは

原点をどこにするのか？ 座標の単位はどうするのか？

図法と投影法



正距方位図法

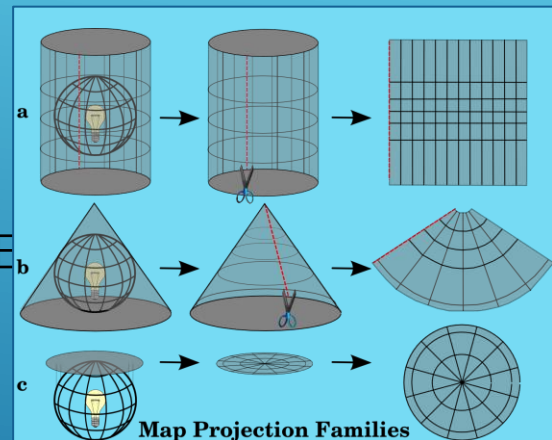
形状・距離・方位・面積等の全てを正確に保つ事は不可能なため、用途に応じて様々な方法・分類がある。

性質で分類

- ↓ 正積図法：面積が正しい
- ↓ 正角図法：任意の二地点間を結んだ直線と経線のなす角度が正
- ↓ 正距図法：距離が正しい

投影で分類

- ↓ 方位図法：地球の周辺に置かれた平面に地球の姿を投影する。
- ↓ 円錐図法：地球に傘のような円錐をかぶせて地球の姿を投影し、平面に切り広げる。
- ↓ 円筒図法：地球に帯を巻くように円筒をかぶせて地球の姿を投影し、平面に切り広げる。



参考：

↓ GIS実習オープン教材

← <https://gis-oer.github.io/gitbook/book/>

↓ オンラインGIS教材

← <https://sites.google.com/view/gis-online-learning/>

↓ 地理教材共有サイト

← <https://sites.google.com/view/geoclass2020/>

はじめてのマップ表示



はじめてのマップ表示

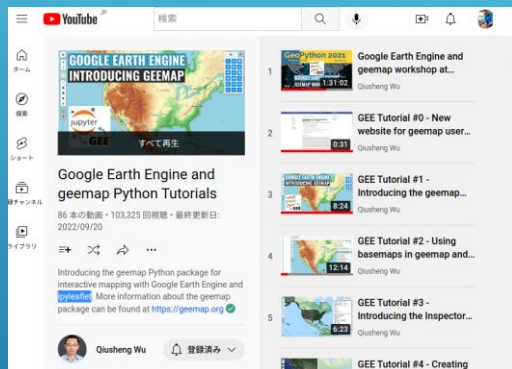
- ↓ geemapとleafmap
- ↓ インストール
- ↓ ベースマップを表示
- ↓ GeoPandas GeoDataFrameデータを表示
- ↓ CSVファイルを表示
- ↓ 2.5D地図で表示

GEEMAPとLEAFMAP

- ↓ インタラクティブなマッピングと地理空間分析を行うための無料でオープンソースのPythonパッケージ
- ↓ ユーザーが最小限のコーディングで地理空間データを分析および視覚化可能にしたもの
- ↓ Google Colab、JupyterNotebook環境で動作
- ↓ テネシー大学 地理学部 准教授のQiusheng Wu, PhDが開発
- ↓ foliumやipyleafletなどのライブラリをベースに構築
- ↓ geemapは、Google Earth Engine (GEE) が動作
- ↓ leafmapは、geemapのスピンオフでgeemapのGEEなしのバージョン

GEEMAPとLEAFMAP

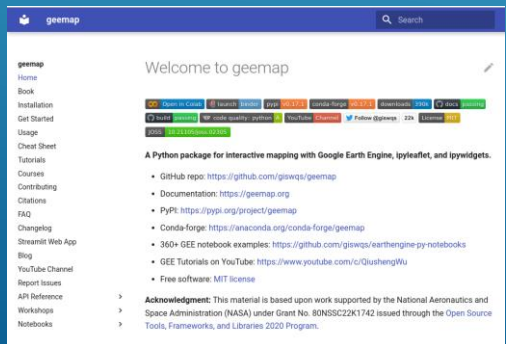
geemap



A Python package for interactive mapping with Google Earth Engine, ipyleaflet, and ipywidgets.

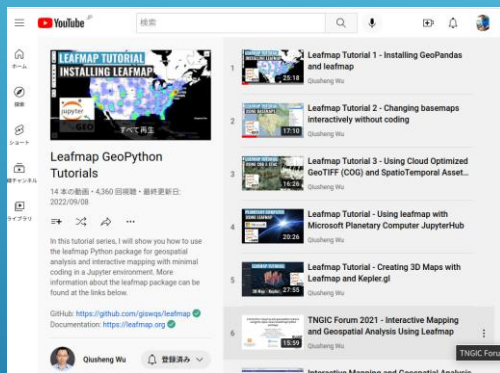
- ↙ GitHub repo: <https://github.com/giswqs/geemap>
- ↙ Documentation: <https://geemap.org>
- ↙ PyPI: <https://pypi.org/project/geemap>
- ↙ Conda-forge: <https://anaconda.org/conda-forge/geemap>
- ↙ 360+ GEE notebook examples: <https://github.com/giswqs/earthengine-py-notebooks>
- ↙ GEE Tutorials on YouTube: <https://www.youtube.com/c/QiushengWu>
- ↙ Free software: MIT license

Google Earth Engine and geemap Python Tutorials
<https://www.youtube.com/playlist?list=PLAxJ4-o7ZoPccOFv1dCwvGI6TYnirRTg3>



GEEMAPとLEAFMAP

leafmap

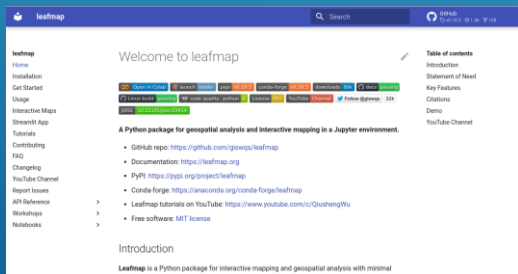


A Python package for geospatial analysis and interactive mapping in a Jupyter environment.

- ↳ GitHub repo: <https://github.com/giswqs/leafmap>
- ↳ Documentation: <https://leafmap.org>
- ↳ PyPI: <https://pypi.org/project/leafmap>
- ↳ Conda-forge: <https://anaconda.org/conda-forge/leafmap>
- ↳ Leafmap tutorials on YouTube: <https://www.youtube.com/c/QiushengWu>
- ↳ Free software: MIT license

Leafmap GeoPython Tutorials

<https://www.youtube.com/playlist?list=PLAxJ4-o7ZoPeMITwB8eyynOG0-CY3CMDw>



インストール

geemap

Install from PyPI

```
pip install geemap
```

Install from conda-forge

```
conda install geemap -c conda-forge
```

依存関係のあるライブラリ(e.g. pygis)をインストール

```
conda create -n gee python
```

```
conda activate gee
```

```
conda install -c conda-forge mamba
```

```
mamba install -c conda-forge geemap pygis
```

インストール

leafmap

Install from PyPI

```
pip install leafmap
```

Install from conda-forge

```
conda install leafmap -c conda-forge
```

依存関係のあるライブラリ(e.g. geopandas, localtileserver, keplergl, pydeck)をインストール

```
conda create -n geo python=3.9
```

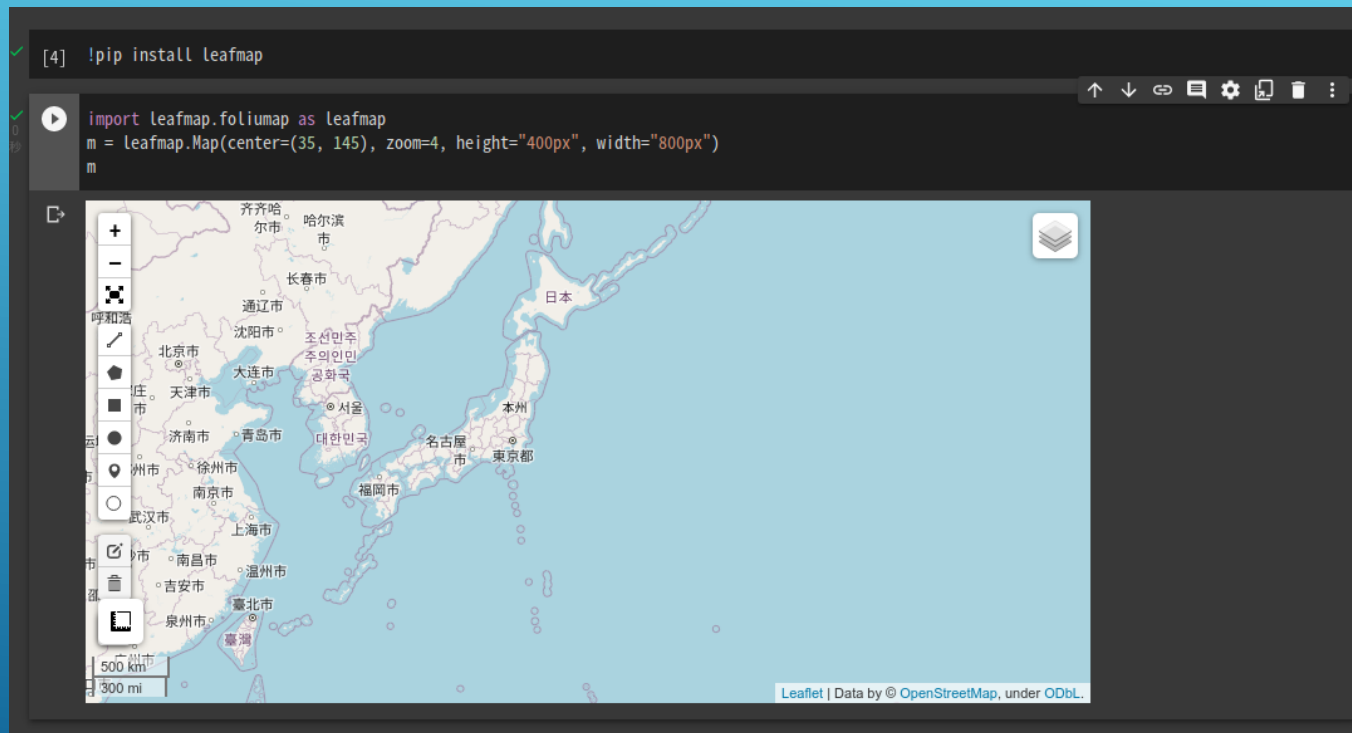
```
conda activate geo
```

```
conda install geopandas
```

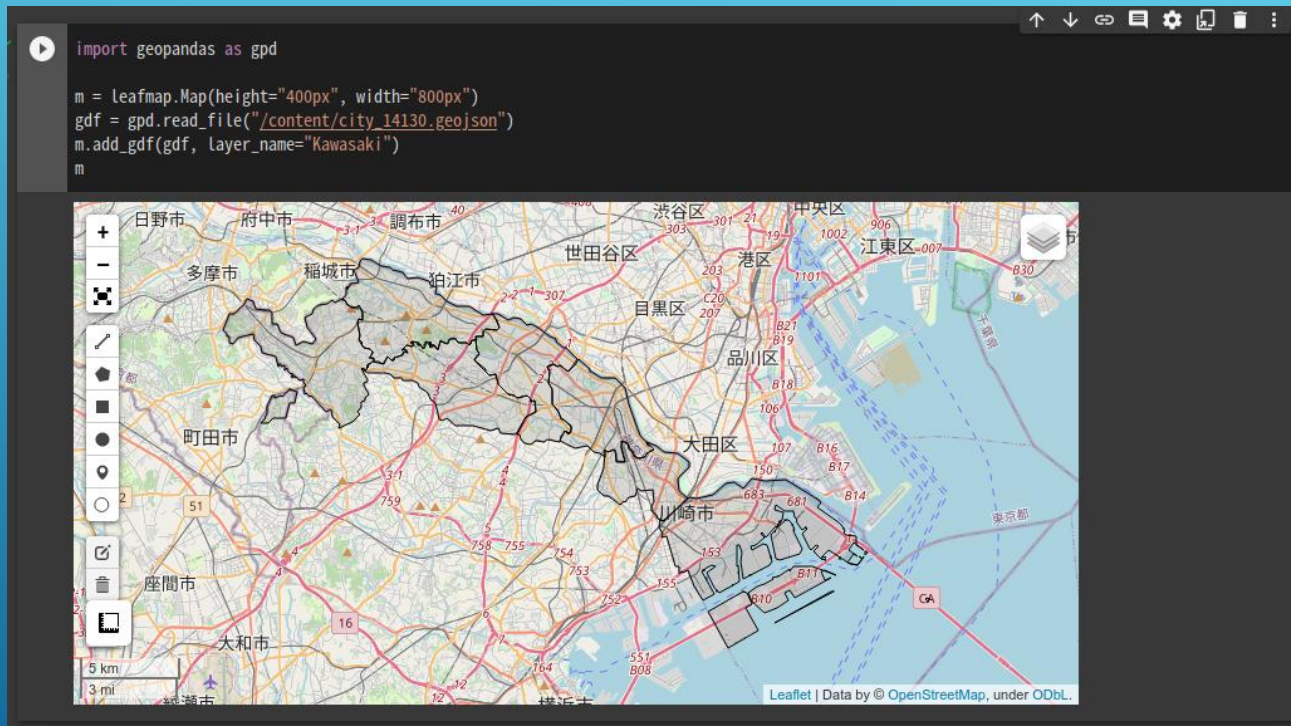
```
conda install mamba -c conda-forge
```

```
mamba install localtileserver keplergl pydeck leafmap -c conda-forge
```

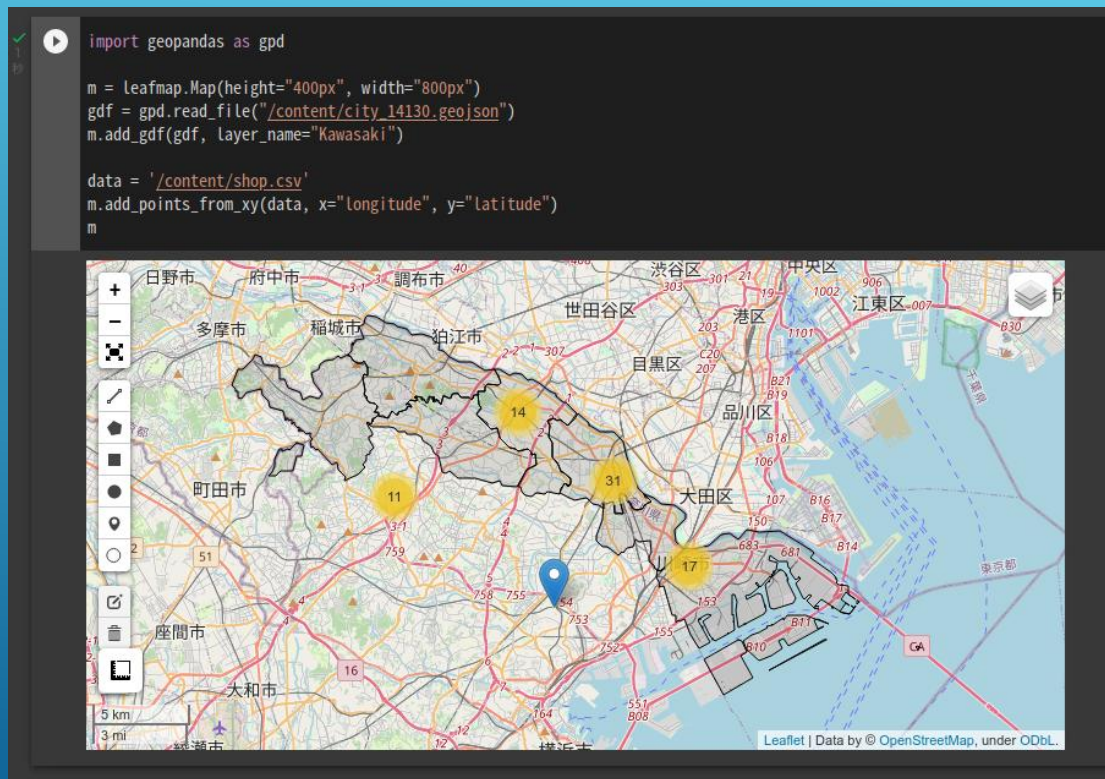
ベースマップを表示



GEOPANDAS GEODATAFRAMEデータを表示

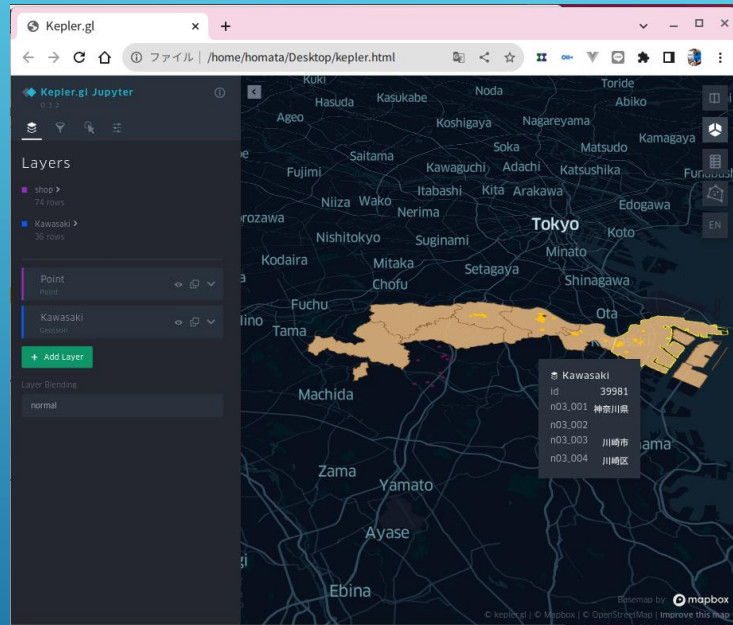
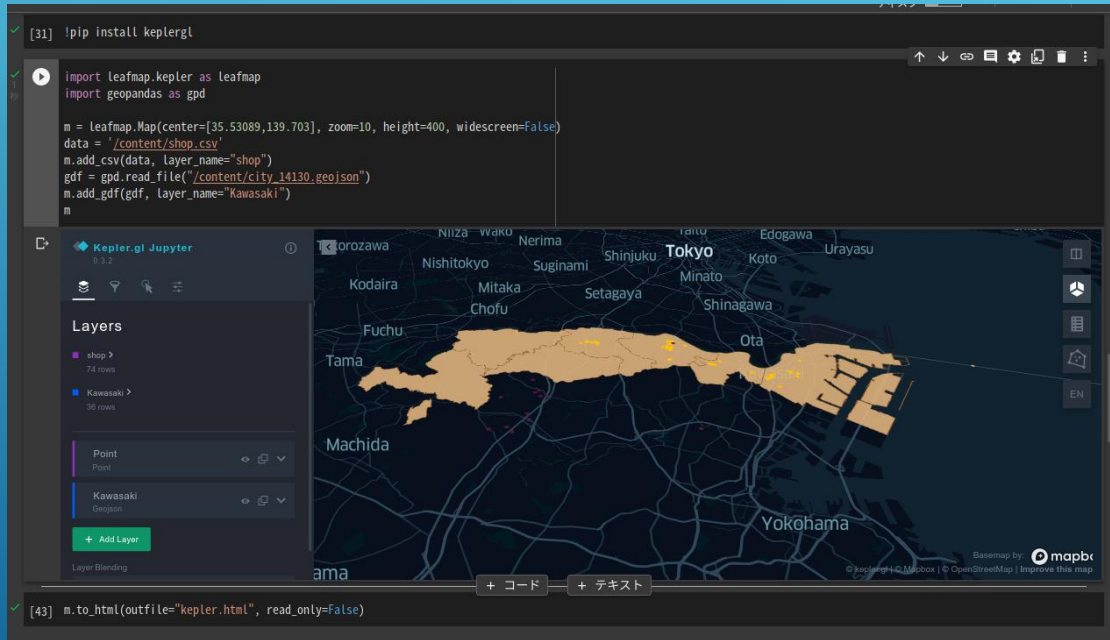


CSVファイルを表示



CSVファイルを読み込んで、マーカーをクラスター表示

2.5D地図で表示




keplerで2.5D表示し、HTMLファイルにエクスポート

はじめての3次元空間データ



はじめての3次元空間データ

- ↓ 点群データとは？
 - ↓ 3次元データのファイル形式
 - ↓ 点群データに便利なツール
 - ↓ 点群データを使った3次元データの可視化
- 
- A series of several parallel white lines of varying lengths, slanted diagonally from the bottom right towards the top right, set against a blue gradient background.

点群データとは？



点群（てんぐん）データは点（ポイント）の集まりのデータであり、直交座標値（ X, Y, Z ）+ 色情報（ R, G, B ）+ α で構成された座標と色情報のある3次元の空間情報データ。ポイントクラウド(point cloud)とも呼ばれている。

既存の画像データやデジタル写真も同じ原理であり、詳細なドットを高密度に集めることで滑らかな質感を再現ができる。

点の密度が高くなれば高精度のデータとなるが、点の密度に比例してデータサイズが膨大となる。

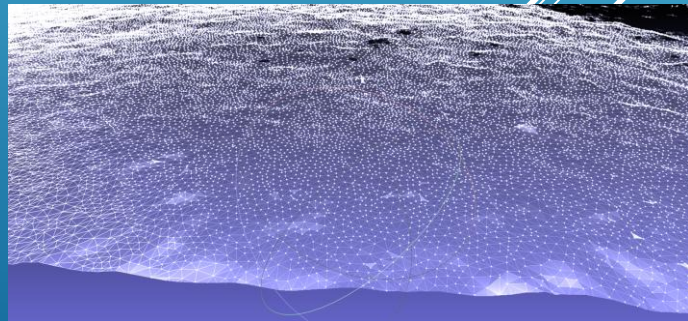
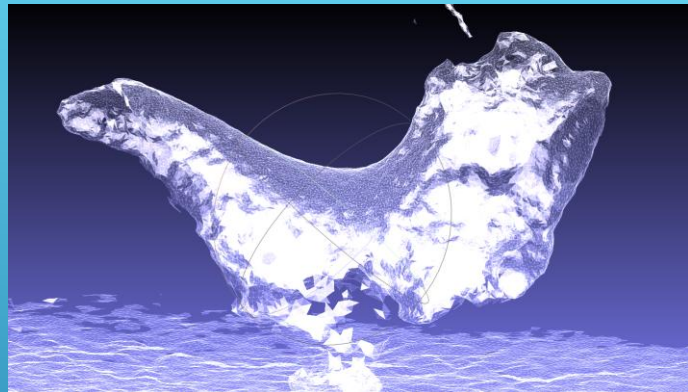
3次元データのファイル形式

↓ 点群形式

- ↪ 点データ
- ↪ 複雑なモデルの形状を正確に表現可能
- ↪ 編集が難しい

↓ メッシュ形式

- ↪ 三角形のポリゴンメッシュ（面形状）のデータ
- ↪ 文字やテクスチャなど面の細かい情報もスキャン可能



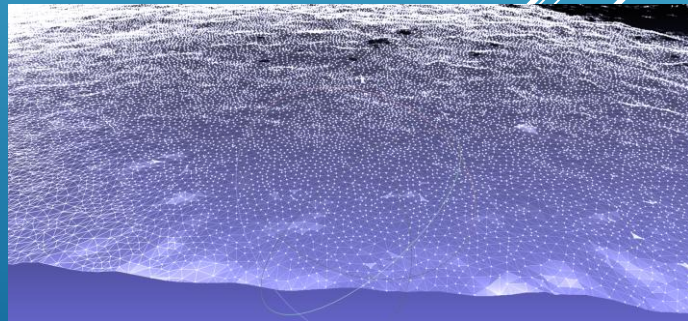
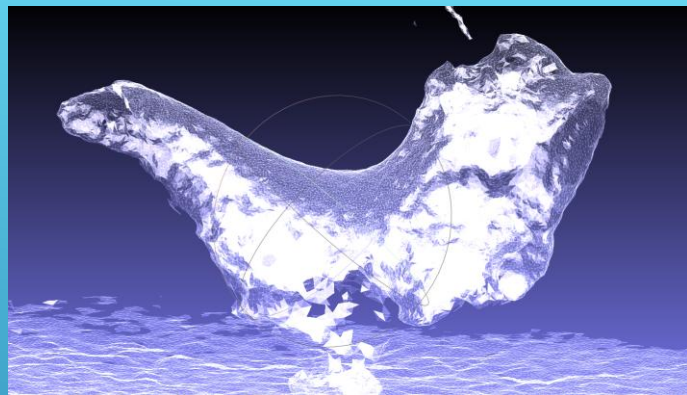
3次元データのファイル形式

↓ 点群形式

- ↪ 点データ
- ↪ 複雑なモデルの形状を正確に表現可能
- ↪ 編集が難しい

↓ メッシュ形式

- ↪ 三角形のポリゴンメッシュ（面形状）のデータ
- ↪ 文字やテクスチャなど面の細かい情報もスキャン可能



3次元データのファイル形式

↓ 点群データ系

- ↪ las、laz形式
- ↪ ply形式 (Polygon File Format)
- ↪ PCD形式 (Point Cloud Data)

↓ メッシュデータ系

- ↪ STL形式 (Standard Triangulated Language)
- ↪ OBJ形式
- ↪ FBX形式

※ 点群データは、las → ply → メッシュデータ（ポリゴン）形式に変換して、使用することが多い
※ 各データのデータ形式は、テキストまたはバイナリデータの形式がある（例 las, ply等）

点群データに便利なツール

CloudCompare

- 点群データの表示、簡易編集に便利
- オープンソースのソフトウェア
- コマンドラインで実行可能
- <https://www.danielgm.net/cc/>

MeshLab

- 点群データやメッシュデータの表示や簡易編集に便利
- オープンソースのソフトウェア
- <https://www.meshlab.net/>

Point Cloud Library

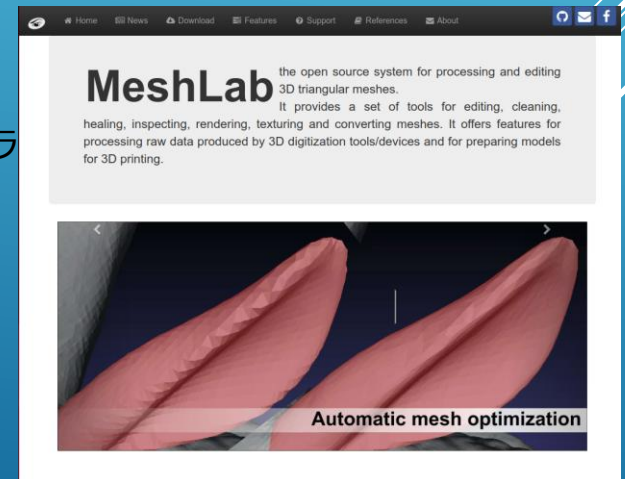
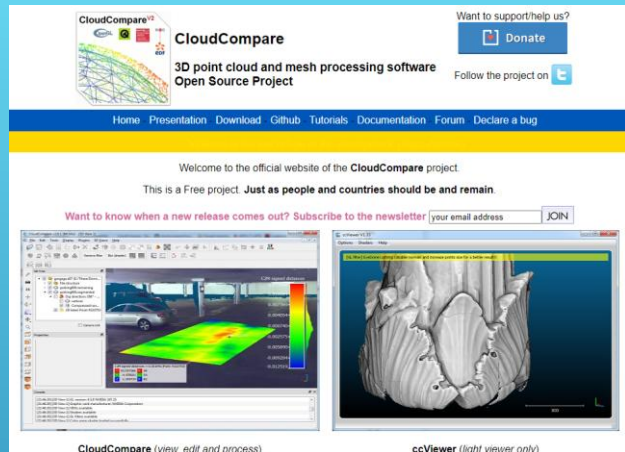
- 3次元の点群データを扱う、C++言語のソフトウェアライブラリ
- オープンソースのソフトウェア

Open3D

- 3次元データを扱うソフトウェアライブラリ
- オープンソースのソフトウェア
- <http://www.open3d.org/>

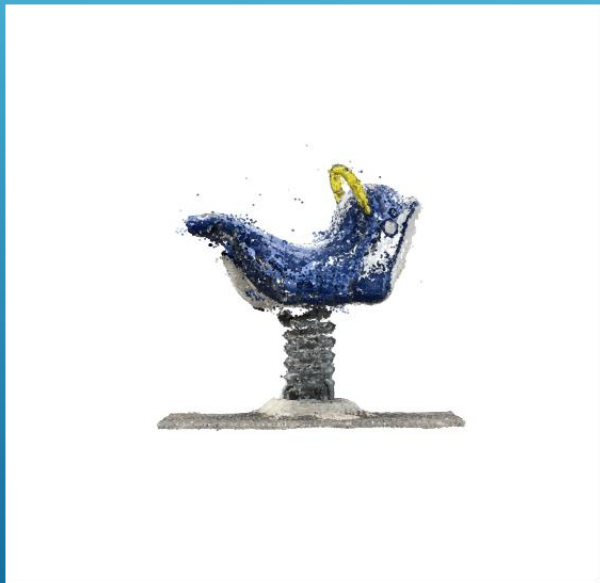
OpenCV

- 画像処理・画像解析のためのライブラリ
- <https://opencv.org/>



点群データを使った3次元データの可視化

点群データ（PLYファイル）をOpen3DとMatplotlibで可視化



Open3Dの「`o3d.visualization.draw_geometries()`」でも3D表示可能
ただし、Google Colabではサポートされていない。
別ウィンドウ表示を`%inline`できれば可能かも

```
[1] !pip install open3d

[3] # Open3DとNumpyをインポートします
import open3d as o3d
import numpy as np

[4] # 点群ファイル (PLY) 読み込み
cloud = o3d.io.read_point_cloud("/content/drive/MyDrive/dataset/image_in/iruka.ply")

[5] # Matplotlibで点群ファイルを表示します
import matplotlib.pyplot as plt
from mpl_toolkits import mplot3d

[6] # 点群データをnp.array形式に変換します
points = np.asarray(cloud.points)

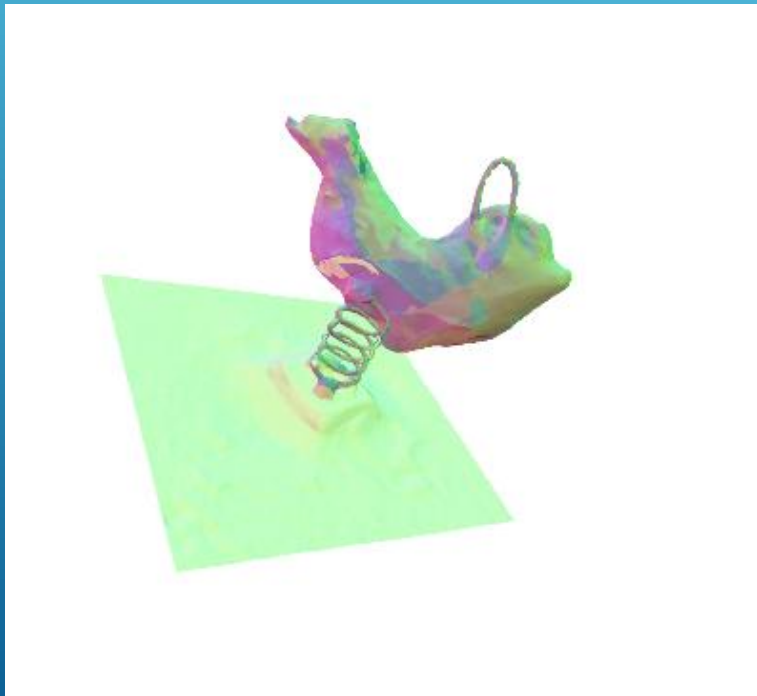
[7] # 色データをnp.array形式に変換します
colors = np.asarray(cloud.colors)

[8] # 表示サイズを指定
import matplotlib.pyplot as plt
from matplotlib.pyplot import figure
plt.rcParams["figure.figsize"] = [10,10] # 図の幅と高さをインチで指定する

[9] # Matplotlibで表示
ax = plt.axes(projection='3d')
ax.view_init(elev=0, azim=0)
ax.axis("off")
ax.scatter(points[:,0], points[:,1], points[:,2], s=1, c=colors)
plt.show()
```


点群データを使った3次元データの可視化

メッシュデータ (OBJファイル) をOpen3dとPlotlyで可視化



```
[ ] # Plotlyをインストール
!pip install plotly

[ ] # メッシュタイプ (OBJファイル) のデータを読み込み
mesh = o3d.io.read_triangle_mesh("/content/drive/MyDrive/dataset/image_in/iruka.obj")

[ ] # 法線を計算
mesh.compute_vertex_normals() # 頂点の法線を計算
mesh.compute_triangle_normals() # 三角形の法線を計算

TriangleMesh with 58260 points and 102006 triangles.

[ ] # 頂点と三角形のデータをnp.array形式に変換
triangles = np.asarray(mesh.triangles)
vertices = np.asarray(mesh.vertices)
# 色を設定
colors = (0.5, 0.5, 0.5) + np.asarray(mesh.triangle_normals) * 0.5 # 三角形の法線データで色を変化させる
colors = tuple(map(tuple, colors))

[ ] # Plotlyをインポート
import plotly.graph_objects as graph_objects

[ ] # Plotlyで表示。出力結果はマウスでインタラクティブに動かすことが可能です
fig = graph_objects.Figure(
    data=[
        graph_objects.Mesh3d(
            x=vertices[:,0], # 頂点のX座標
            y=vertices[:,1], # 頂点のY座標
            z=vertices[:,2], # 頂点のZ座標
            i=triangles[:,0], # 三角形の1番目の座標
            j=triangles[:,1], # 三角形の2番目の座標
            k=triangles[:,2], # 三角形の3番目の座標
            facecolor=colors, # 面の色を設定
            opacity=0.50) # 表面の不透明度を設定
    ],
    layout=dict(
        scene=dict(
            xaxis=dict(visible=False), # X軸非表示
            yaxis=dict(visible=False), # Y軸非表示
            zaxis=dict(visible=False) # Z軸非表示
        )
    )
)
fig.show() # 表示
```

Scaniverse (スキャニバース)

Free 3D scanner

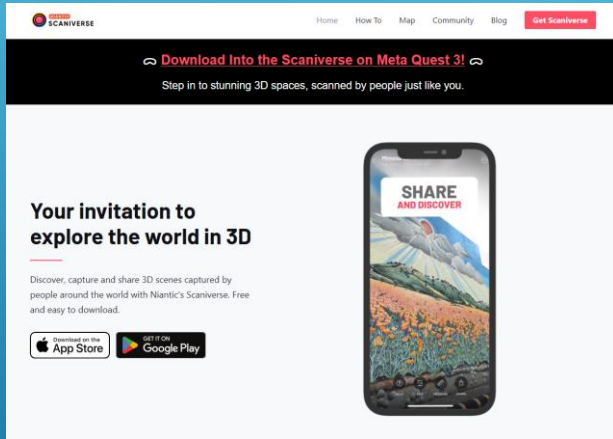
Scaniverseは、Niantic社が開発したスマートフォンカメラやLiDARセンサーを使って、オブジェクトや空間を3Dモデルとしてスキャン・編集・共有できる無料アプリ

LiDARを搭載していないデバイスでもカメラ機能のみでスキャン可能なほか、高品質な3D Gaussian Splatting (3DGS) モードにより、光沢のあるものや複雑な形状もリアルにスキャンできる

作成した3Dモデルは、OBJやFBXなどの形式でエクスポートして他の3Dソフトで利用したり、InstagramなどのSNSで共有したりすることができる

Niantic(ナイアンティック)はポケモンGOを開発してる会社で、2010年に設立されたにGoogleの社内スタートアップ

Google Earthの元となる製品を開発していた



SCANIVERSE

<https://scaniverse.com/>



地理空間情報データとデータ活用事例

Several white lines of varying lengths and angles are positioned in the bottom right corner of the slide, creating a modern, geometric design element.

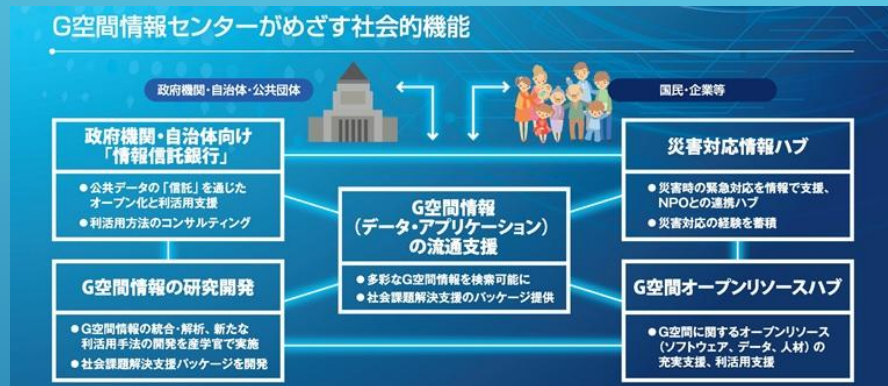
地理空間情報データとデータ活用事例

地理空間情報データとデータ活用事例を数例紹介

- ↓ G空間情報センター
 - ↓ Project PLATEAU
 - ↓ 国土交通データプラットフォーム
 - ↓ My City Construction
 - ↓ My City Forecast
 - ↓ デジタルシティサービス
 - ↓ My City Report
- 

G空間情報センター

地理空間情報（＝G空間情報）の有効活用と流通促進を行うためのデータ流通支援プラットフォーム



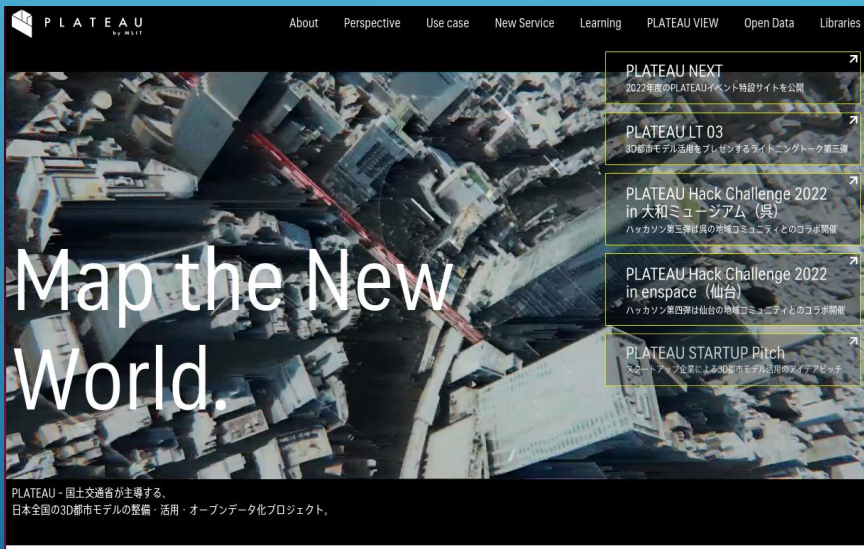
G空間情報センターは、産官学の様々な機関が保有する地理空間情報を円滑に流通し、社会的な価値を生み出すことを支援する機関です。

平成24年3月に政府で閣議決定された地理空間情報活用推進基本計画に基づき、設立され、一般社団法人社会基盤情報流通推進協議会が運用を行っている

<https://front.geospatial.jp>

PROJECT PLATEAU (プラトー)

国土交通省が進める3D都市モデルの整備・活用・オープンデータ化推進するプロジェクト



<https://www.mlit.go.jp/plateau/>

PLATEAUデータは、CityGML形式でデータを整備

データソース

1. 都市計画基本図（基盤地図情報）
2. 都市計画基礎調査
3. 公共測量成果（航空写真又はLP）

・データ特性

- 国際標準化団体OGCが策定した「CityGML 2.0」を採用
- ・高い拡張性・互換性
国際標準規格であるため、BIMなどの多様な分野のデータとの連携・交換やソフトウェア対応が可能
日本独自の情報を追加可能
- ・オープンデータによるイノベーション創出
二次利用等が可能なオープンライセンスを採用し
「G空間情報センター」で一般に公開

CityGML形式は中間フォーマットとして、利用者が「使いやすいフォーマット」に変換した上で利用することを想定

国土交通データプラットフォーム

国・地方自治体の保有する橋梁やトンネル、ダムや水門などの社会インフラ（施設）データの整備・活用・オープンデータ化推進するプロジェクト



国土交通省が運用しているデータプラットフォーム

国土交通省は国土や都市、交通、気象等の多くのデータを保有している機関であるものの、データは個々の部局ごとに管理されているものが多い。多く保有するデータと民間等のデータを連携し、フィジカル（現実）空間の事象をサイバー空間に再現するデジタルツインにより、業務の効率化やスマートシティ等の国土交通省の施策の高度化、産学官連携によるイノベーションの創出を目指したもの

MY CITY CONSTRUCTION

電子納品成果の包括的な活用と共有が可能なオンライン型電子納品システム

My City Construction

ログイン 組織ユーザー作成申請

業務・工事データ一覧 地図検索 納品料金 マニュアルview 官民の取組状況 お問い合わせ 統計 施設検索



オンライン電子納品とは？ 2020年度の活動内容はこちら。2019年度以前の活動内容はこちら。

MCCの概要

MCCの運営主体

MCCの利用効果【発注者・受注者】

近年、国土交通省における「i-Construction」などの施策が本格的に始まったことを背景に、地方自治体などの公共工事の現場に関わって、点群データやUAVなどによる撮影データ等からなる三次元データがますます取得されていくと考えられています。一方、こうしたデータの利活用は始まったばかりであるとともに、これまで行われてきた電子納品成果についても、例えば新規建設工事と維持修繕工事といった異なる工事種でのデータ利用についてもまだ進んでいない現状にあります。その理由として、今までの電子納品成果が発注者側の電子納品保管管理システムにきちんと登録されなかったり、DVDやブルーレイといった電子媒体単位で管理され、これらの蓄積されたデータを円滑に活用することが困難な状況にあると考えられます。

そこで本システムは、受注者が検査前に電子納品成果をアップロードすることで、点群データやUAV撮影データ等の重いデータを円滑にプレビュー表示・検索することや、オープンデータについてはG空間情報センターからも検索可能な、自治体単位でも採用しやすい低コストかつ包括的なオンライン型の電子納品システム（My City Construction）を設計・構築することを目指しています。

本システムの開発ならびに実証実験にあたっては、国土交通省建設技術研究開発助成制度（平成29-30年度）の支援を受け、東京大学生産技術研究所（研究代表：関本義秀研究室）、株式会社建設技術研究所、（一社）社会基盤情報流通推進協議会の3者が主体となり行われ、その後、令和2年度より本格運用に入っています。

オンライン電子納品実施数 *1

実施受注法人数 *2

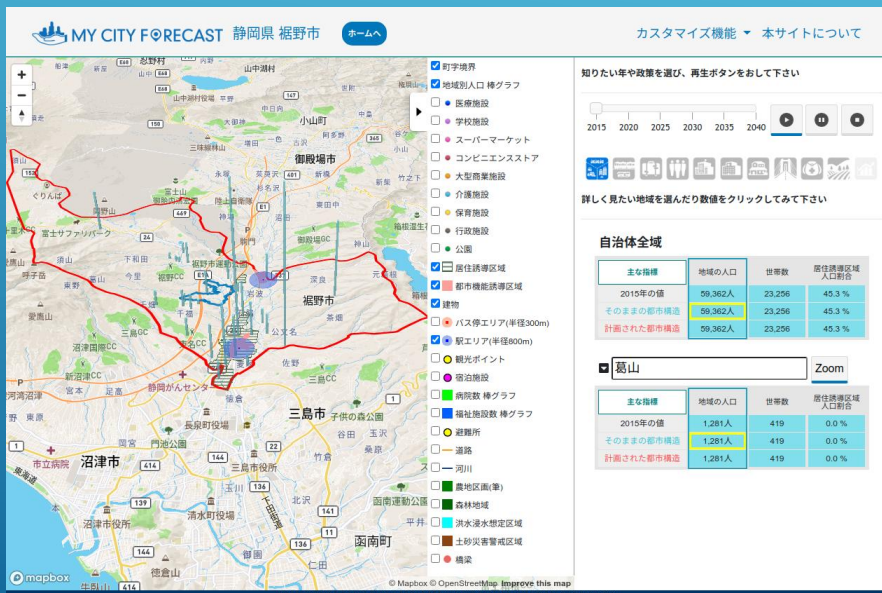
実施自治体数 *2

受注者が検査前に電子納品成果をアップロードすることで、点群データやUAV撮影データ等の重いデータを円滑にプレビュー表示・検索することや、オープンデータについてはG空間情報センターからも検索可能な、自治体単位でも採用しやすい低コストかつ包括的なオンライン型電子納品システム（仮称：My City Construction）を設計・構築した

<https://mycityconstruction.jp/>

MY CITY FORECAST VER2.0

現状の人口分布・施設配置データをもとに、2015年～2040年に想定される居住地域の環境を可視化



将来の地域の世帯・人口データを推計し、指標化

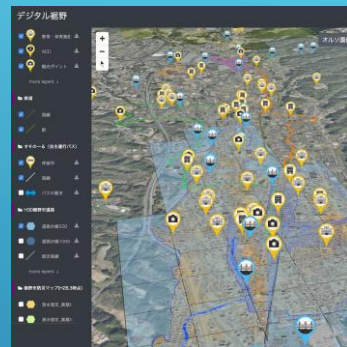
My City Forecast Ver2.0 コンセプト

- ↓ 世帯をエージェントとした世帯推計データを開発し、自治体単位の転出入等も加味した確率で考慮した2020年～2040年の世帯推移モデルを構築
- ↓ 小地域をベースに可視化や集計値算出を行い、よりミクロなスケールでリアリティが伝わるように改良
- ↓ 従来の14指標から政策分野ごとに10テーマ（政策）に分類し直し、各分野での重要指標も可視化できるようにした

<https://mycityforecast.ne>

デジタルシティサービス

リアルタイムなデジタルツイン環境を提供していくためのデータセットそのものと三次元都市空間の可視化環境



リアルタイムなデジタルツイン環境を提供していくために、データセットそのものと三次元都市空間の可視化環境を表裏一体的に扱う、ホットスタンプ型のデジタルシティ環境を低廉な価格で地方公共団体向けに提供している。

具体的には、オリジナルデータを一度、データレジストリ上に登録するとスムーズに可視化画像を構築し、三次元都市空間上でもスムーズに可視化が可能となる

例えば、静岡県裾野市、富山県南砺市では、2019年度に本サービスを用いたデジタル裾野、デジタル南砺を試験的に構築し、2020年4月に公開するとともに、分野ごとに市民協働で課題解決を行っていく

<https://www.digitalsmartcity.jp/>

MY CITY REPORT

市民協働投稿サービス / 道路損傷検出サービス

My City Report for Citizens 市民協働投稿サービス

My City Report for citizens は、市民と自治体が協働してまちの課題に取り組むことができる市民協働投稿サービスです。



My City Reportコンソーシアム事務局

- 一般社団法人社会基盤情報流通推進協議会
- 合同会社Georepublic Japan
- 株式会社アーバンエクステクノロジーズ



My City Report for road managers 道路損傷検出サービス

My City Report for road managers(は、これまで多くの自治体で行われていた、車上からの目視による道路損傷確認業務をAI(人工知能)に置き換え、正確かつスピーディな道路損傷個所検出を行い、業務の効率化を図るサービス



<https://www.mycityreport.jp/>