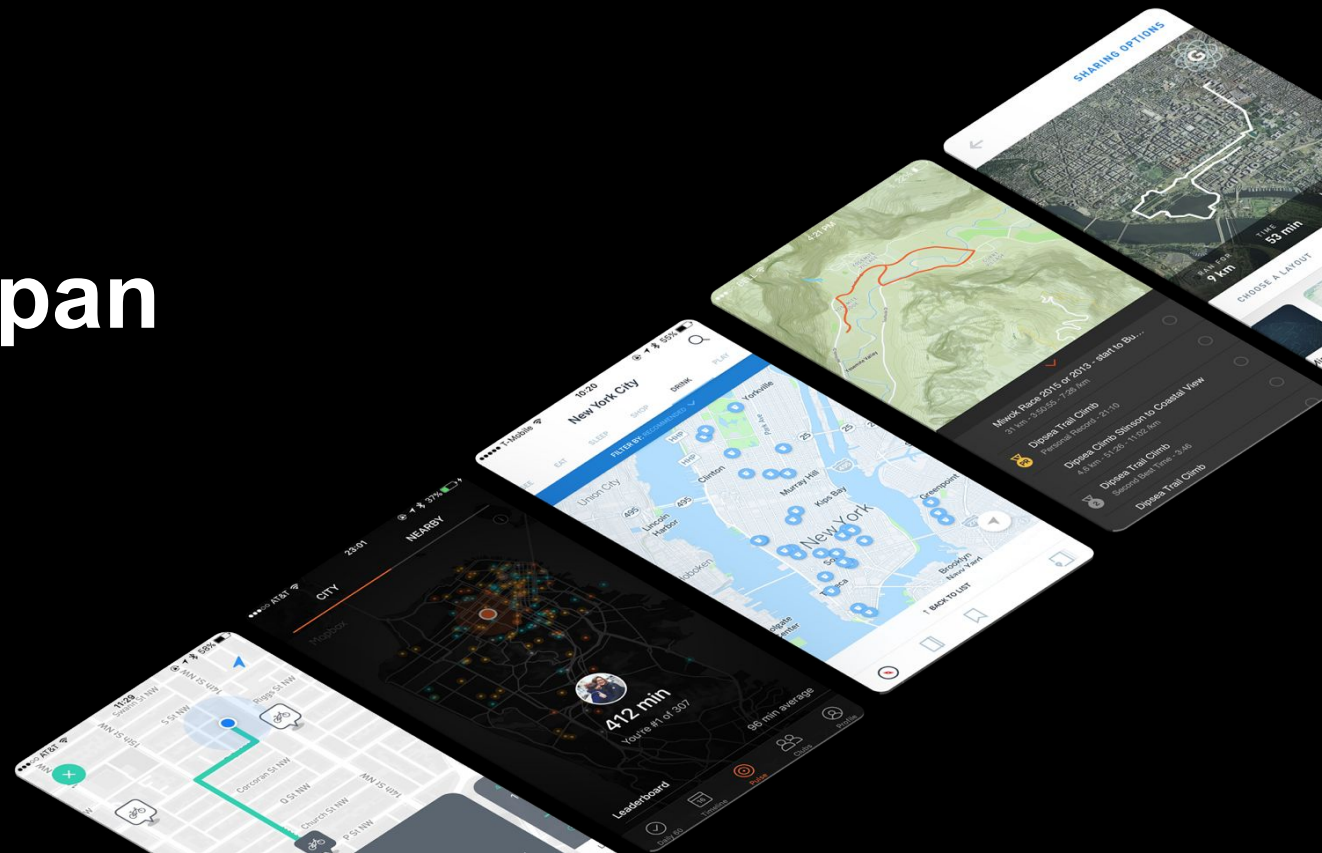
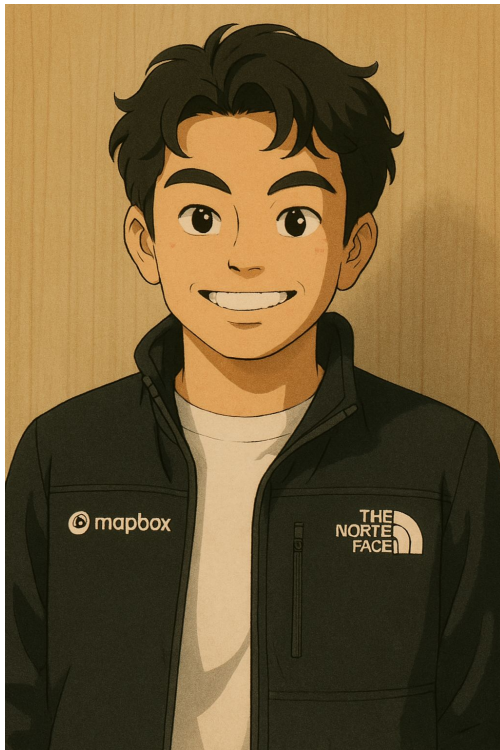




Mobility Night #3 Mapbox Japan





日向野 邦春

Mapbox Japan/ Account Manager

経歴

学生時代にGIS(OSM)に興味を持ち、2021年にmapboxでのインターンを開始、2023年4月にmapboxにAccount Managerとして入社。

担当業界

物流・タクシー・ライドシェア



LinkedIn



sansan

マップボックス 会社概要

クリエイターとエンジニアのための地図開発プラットフォーム

社名	Mapbox Inc.
代表	ピーター・シロタ
設立年	2010年
従業員数	約700名
Web	https://www.mapbox.com/
主な資金 調達先	ソフトバンクグループソフトバンクビ ジョンファンド

エンジニア 中心文化

グローバル700人以上
の従業員のうち
過半がエンジニア

技術系 ユニコーン企業

評価額1,000億円超
ビジョンファンドの出資

Mapbox solutions are used across various industries

Automotive



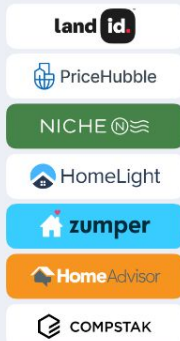
BI & Analytics



Media



Real Estate



Travel



Telecom



Weather



Recreation



Retail



Government



Consumer Apps



On-Demand Logistics

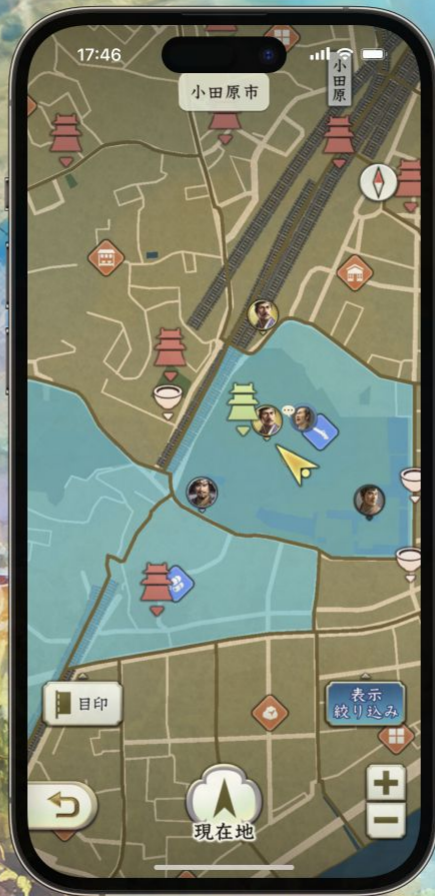


Energy & Agriculture





The new Electric Mini





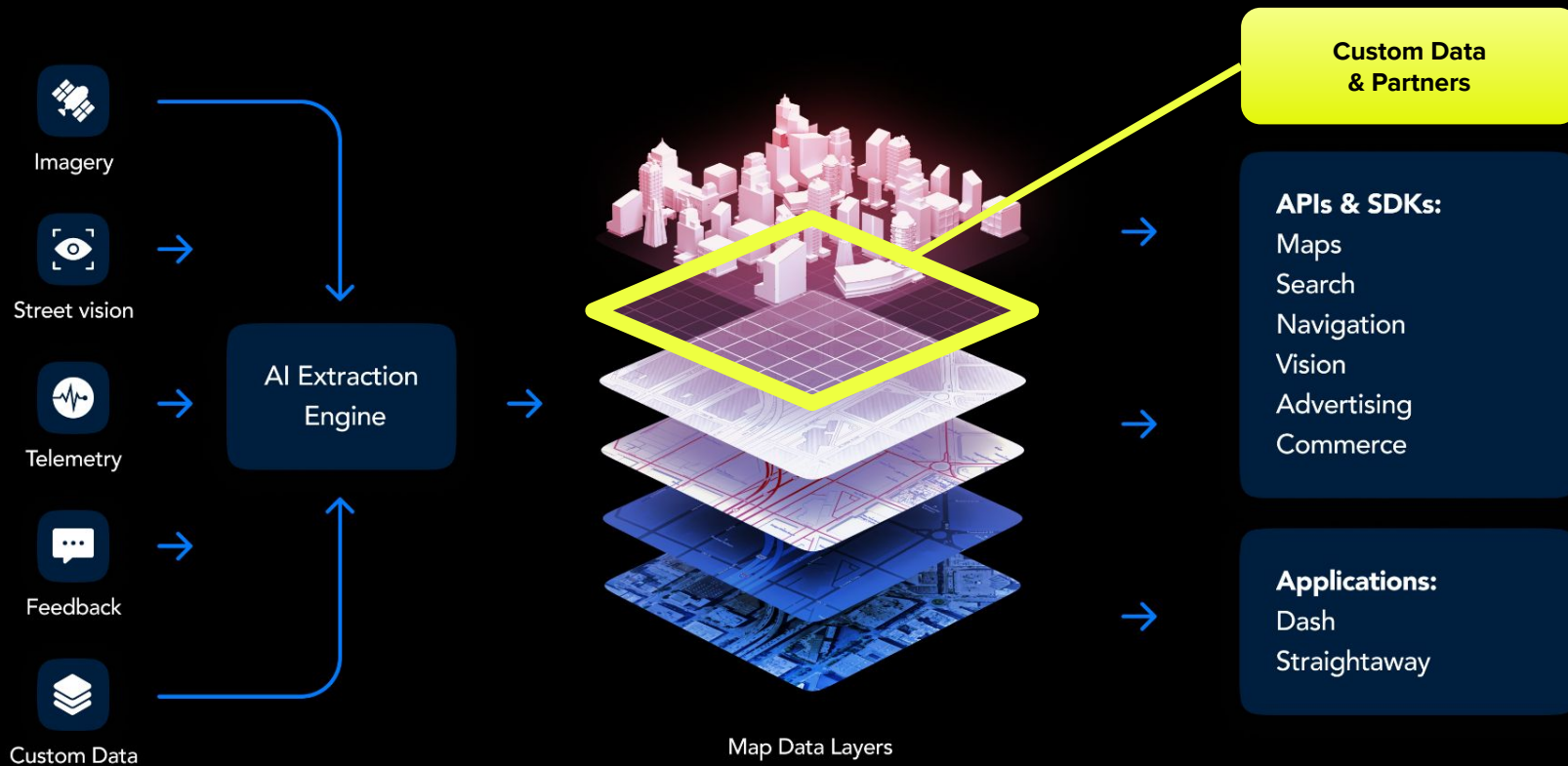
400万人以上
登録開発者

7億人以上
月間アクティブユーザー

34億km以上
1週間で扱う道路データ

70万件以上
1週間の地図更新

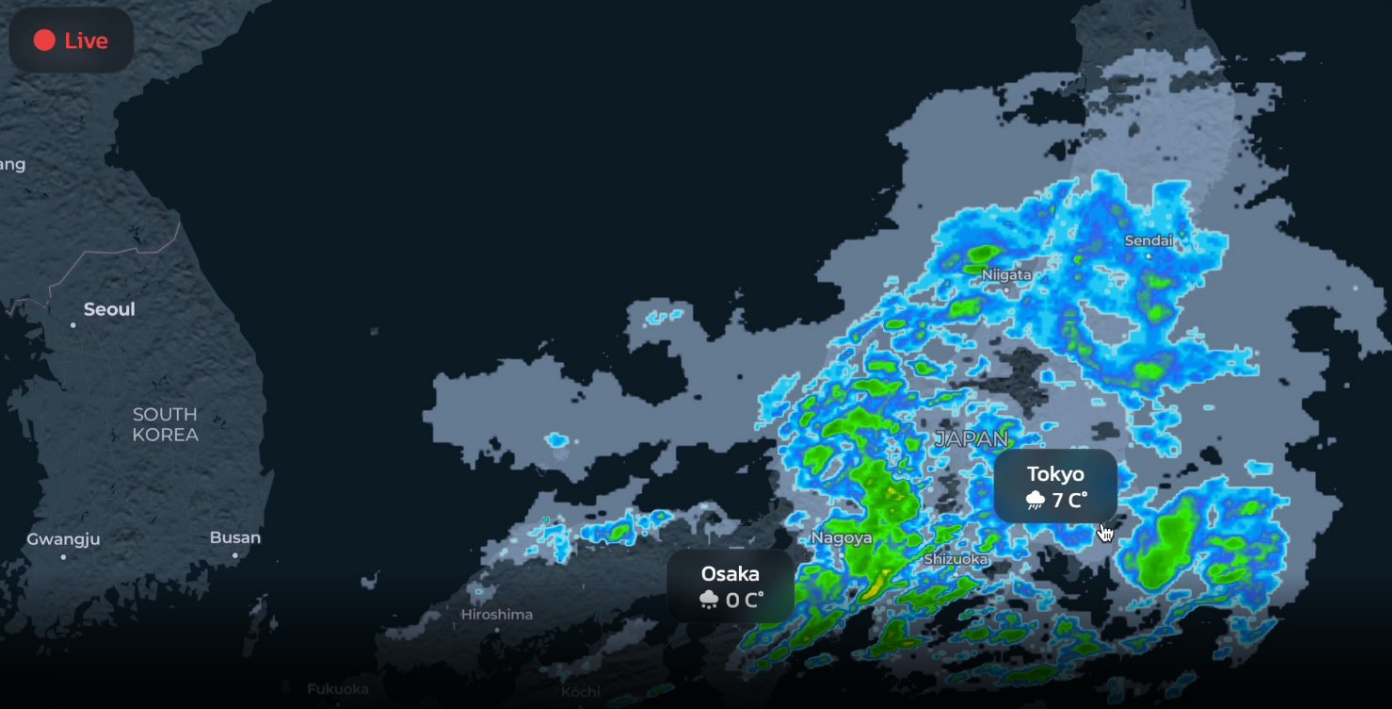
Mapbox Intelligent Live Location Platform



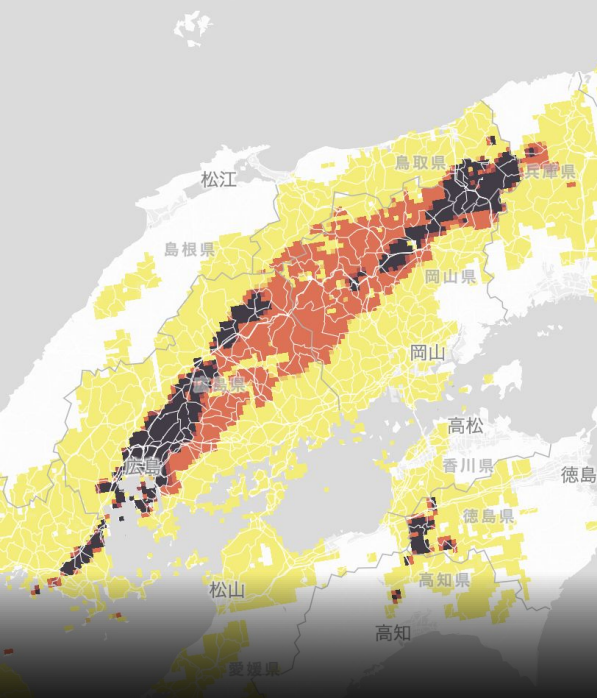
A Mapbox building block for every purpose



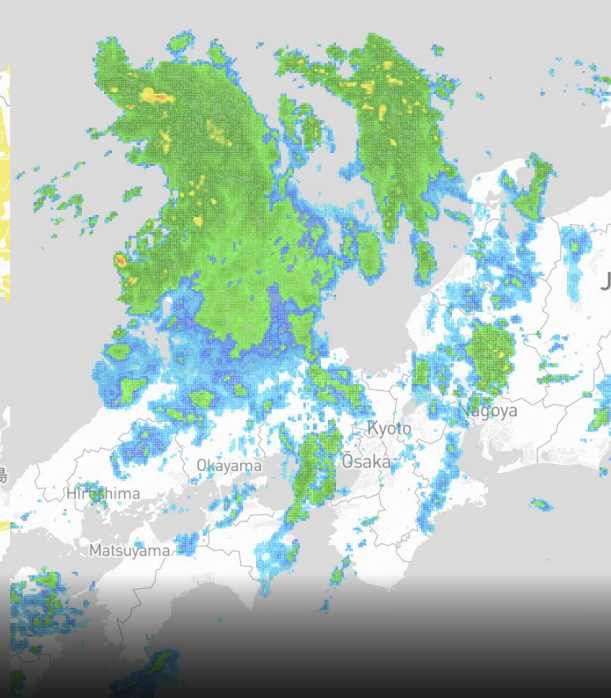
● Live



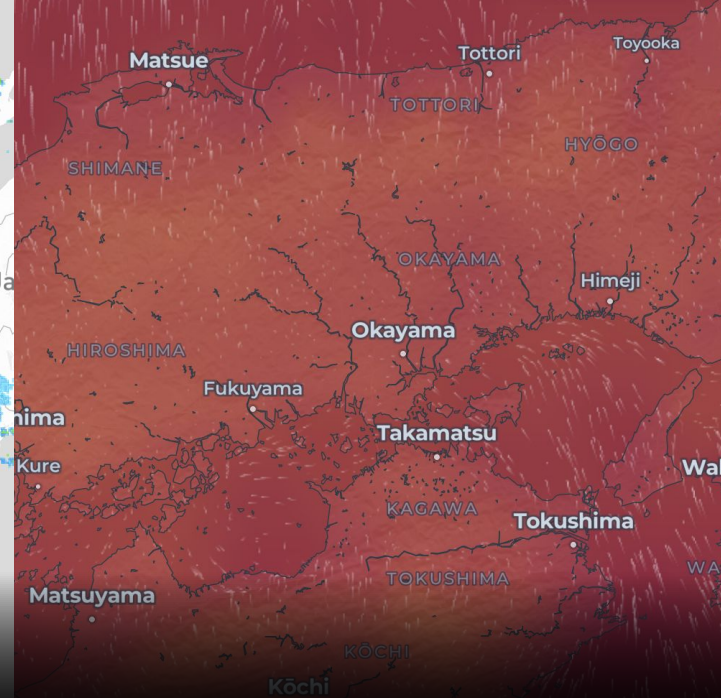
Japan Weather Layer



土砂災害リスク



降水量実況・予測



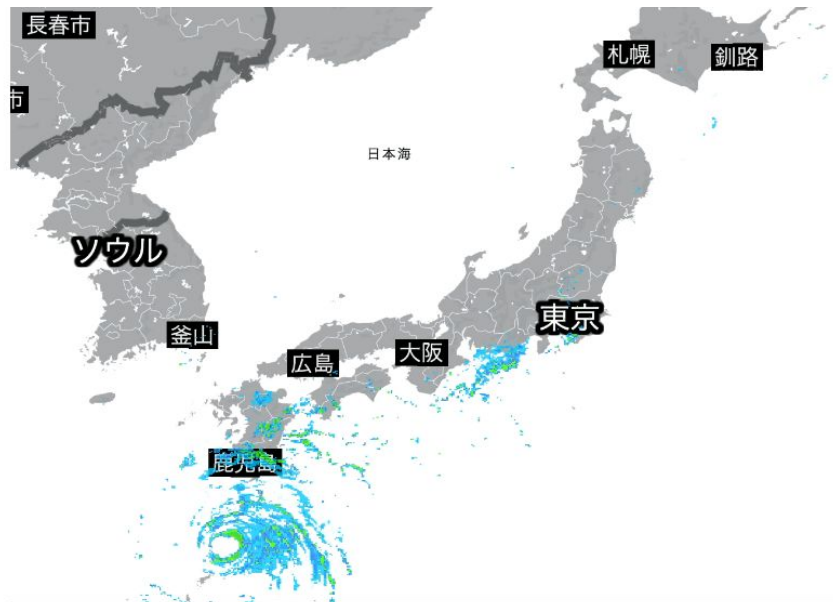
風と気温の数値予報

様々な気象データをマップボックス地図に重畳可能

自分の今いる位置と、そこにおける気象の状況(降雨、落雷、風速など)を直感的に把握できるため、気象と地図は相性抜群。特に他国と比べ、日本では雨雲レーダーが日常的なツールとなっている。

国内での著名な 気象x地図のサービス

- ・Yahoo! 天気
- ・ウェザーニューズ
- ・Tenki.jp
- ・気象庁
- ・NERV (特務機関 NERV 防災アプリ)
- ・NHK

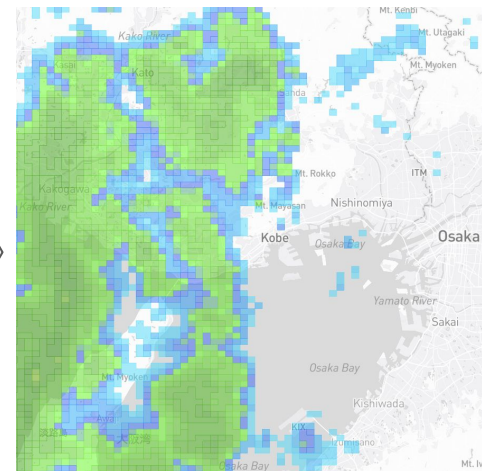


(注) 上記はMapboxによる雨雲レーダーデモです。

日本の気象データは、希望すれば有償で気象庁から購入が可能。但し、GRIB2(これ自体は国際規格)というバイナリ形式になっており、そのままでは地図に重畳できない。

```
1.24819:0:grid_template=0:winds(N/S):
  lat-lon grid:(40 x 40) units 1e-06 input WE:NS output WE:SN
res 48
  lat 21.329166 to 21.004166 by 0.008333
  lon 123.006250 to 123.493750 by 0.012500 #points=1600
check_pdt_size: pdt=50011 needs to be added to pdt_len(..)
1.24820:0:grid_template=0:winds(N/S):
  lat-lon grid:(40 x 40) units 1e-06 input WE:NS output WE:SN
res 48
  lat 21.329166 to 21.004166 by 0.008333
  lon 123.506250 to 123.993750 by 0.012500 #points=1600
check_pdt_size: pdt=50011 needs to be added to pdt_len(..)
1.24821:0:grid_template=0:winds(N/S):
  lat-lon grid:(40 x 40) units 1e-06 input WE:NS output WE:SN
res 48
  lat 21.329166 to 21.004166 by 0.008333
  lon 124.006250 to 124.493750 by 0.012500 #points=1600
check_pdt_size: pdt=50011 needs to be added to pdt_len(..)
1.24822:0:grid_template=0:winds(N/S):
  lat-lon grid:(40 x 40) units 1e-06 input WE:NS output WE:SN
res 48
  lat 21.329166 to 21.004166 by 0.008333
  lon 124.506250 to 124.993750 by 0.012500 #points=1600
```

```
{
  "bbox": [143.5, 46.324999999999996, 143.5125, 46.33333333333333], "geometry":
  {
    "coordinates": [[[143.5, 46.33333333333333], [143.5, 46.324999999999996], [143.5125,
    46.324999999999996], [143.5125, 46.33333333333333], [143.5, 46.33333333333333]]], "type":
    "Polygon"}, {"id": "merged.tif:0", "properties": {"filename": "merged.tif", "val": 2.0},
    "type": "Feature"}
  ],
  "bbox": [143.5125, 46.324999999999996, 143.525, 46.33333333333333], "geometry":
  {
    "coordinates": [[[143.5125, 46.33333333333333], [143.5125, 46.324999999999996], [143.525,
    46.324999999999996], [143.525, 46.33333333333333], [143.5125, 46.33333333333333]]], "type":
    "Polygon"}, {"id": "merged.tif:1", "properties": {"filename": "merged.tif", "val": 3.0},
    "type": "Feature"}
  ],
  "bbox": [143.525, 46.324999999999996, 143.5375, 46.33333333333333], "geometry":
  {
    "coordinates": [[[143.525, 46.33333333333333], [143.525, 46.324999999999996], [143.5375,
    46.324999999999996], [143.5375, 46.33333333333333], [143.525, 46.33333333333333]]], "type":
    "Polygon"}, {"id": "merged.tif:2", "properties": {"filename": "merged.tif", "val": 4.0},
    "type": "Feature"}
  ],
  "bbox": [143.5375, 46.324999999999996, 143.55, 46.33333333333333], "geometry":
  {
    "coordinates": [[[143.5375, 46.33333333333333], [143.5375, 46.324999999999996], [143.55,
    46.324999999999996], [143.55, 46.33333333333333], [143.5375, 46.33333333333333]]], "type":
    "Polygon"}, {"id": "merged.tif:3", "properties": {"filename": "merged.tif", "val": 6.0},
    "type": "Feature"}
  ],
  "bbox": [143.55, 46.324999999999996, 143.5625, 46.33333333333333], "geometry":
  {
    "coordinates": [[[143.55, 46.33333333333333], [143.55, 46.324999999999996], [143.5625,
    46.324999999999996], [143.5625, 46.33333333333333], [143.55, 46.33333333333333]]], "type":
    "Polygon"}, {"id": "merged.tif:4", "properties": {"filename": "merged.tif", "val": 7.0},
    "type": "Feature"}
  ]
}
```



GRIB(データ構造等..気象データにより異なる)を解析

geojson形式に変換

地図へ重畳

オプション:MTS等で
タイルセットに変換

Mapboxユーザーの方であれば、タイルセットHDを指定する + シンプルな実装で気象データを地図に重畳できます。金額は、リクエストされたタイルの数により決まります。

従来のフロー：コストもノウハウも時間も必要

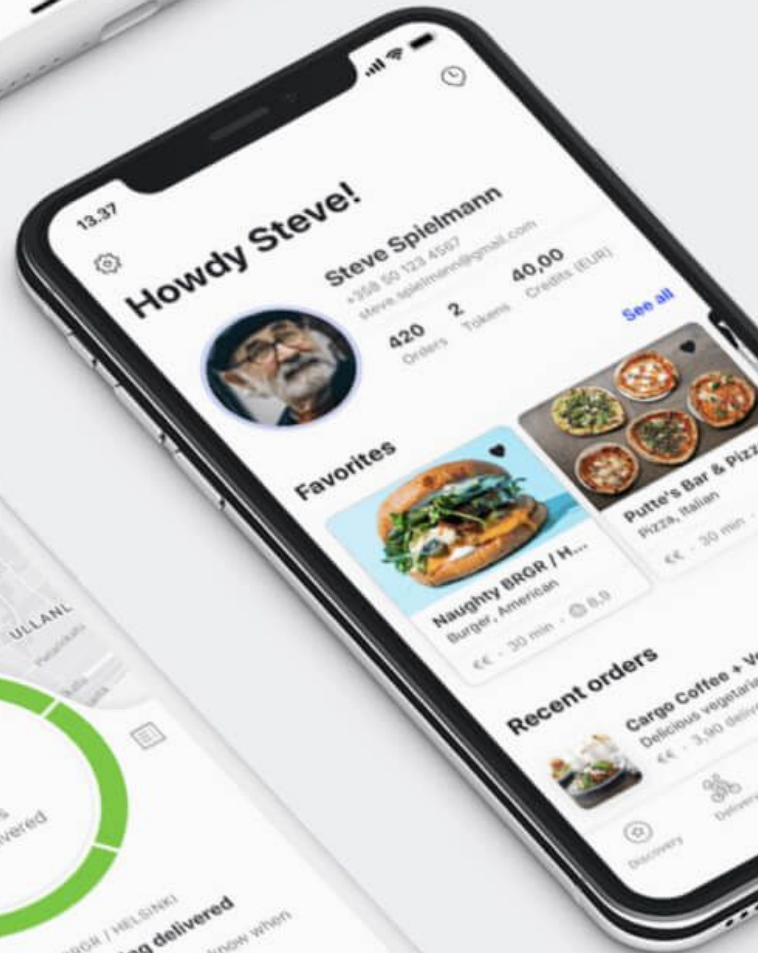
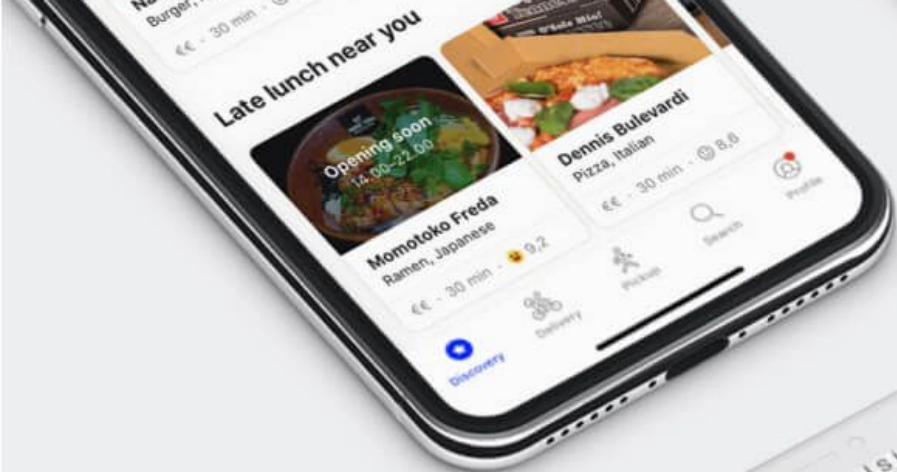


不要! (Japan Weather Serviceで既に対応済みのため)

→ フロント側の開発も Mapboxがマニュアル提供で実装をサポート

プライベート β: 2025年6月から開始します！ご興味のある方はご連絡ください。





STEP1: まずはMapboxにご連絡ください

STEP2: Japan Weather Layerを貴社Mapboxアカウントにて有効化

-- 以降、貴社作業 ---

STEP3: Mapbox Studioにて対象のタイルセットを確認

STEP4: MapsSDKまたは、GLJSを使い、貴社システム(フロントエンド)にて当該タイルセットの参照を実装

STEP5: 続いて、スタイリング(色、透明度など)を実装

-- 以降、オプション(気象データにより必須) ---

Option: (風、波など)風のアニメーションを設定 *MapsSDK/GLJSの機能として提供

※詳細なマニュアルは別途公開予定。本日はあくまでイメージを持っていただくためのご紹介です。

STEP1: まずはMapboxにご連絡ください

STEP2: Japan Weather Layerを貴社Mapboxアカウントにて有効化

-- 以降、貴社作業 ---

STEP3: Mapbox Studioにて対象のタイルセットを確認

STEP4: MapsSDKまたは、GLJSを使い、貴社システム(フロントエンド)にて当該タイルセットの参照を実装

STEP5: 続いて、スタイリング(色、透明度など)を実装

-- 以降、オプション(気象データにより必須) ---

Option: (風、波など)風のアニメーションを設定 *MapsSDK/GLJSの機能として提供

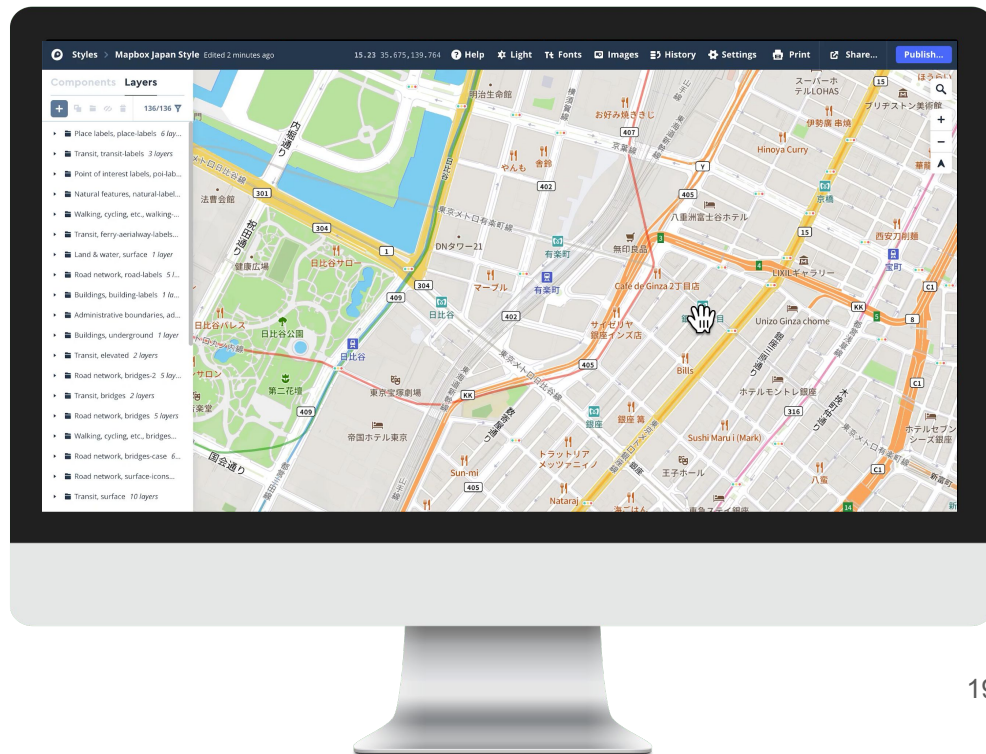
Mapbox Studio: デザインツール

使いやすい開発環境で独自の地図デザイン作成が可能に

- 豊富なStyle Template
- コード不要のデザイン
- 地図データの色やフォントの変更
- 各レイヤー毎の細かな制御
- ズームレベルに応じたスタイル調整
- お客様の持つ動的データの実装

操作イメージ:

<https://apps.mapbox.com/cartogram/#13.01/40.7251/-74.0051>



svc-okta-mapbox-staf...

Home

Tools

Style editor

Data manager

Directions Debug

Data Explorer

Admin

Tokens

Invoices

Statistics

Settings

Admin

Help & resources

Tilesets

Datasets

Search

0 selected

Browse: Custom

Sort by: Date edited

1-25 of 347 files

Trash

GFS kenji-shima.fcst-20240620203000-6h-s-nied

Modified 10 days ago • Private

GFS kenji-shima.nowcast-20240620195000-sel-nied

Modified 10 days ago • Private

kenji-shima.forecast-sel-20240620200000

Modified 10 days ago • Private

kenji-shima.e90m-select-20240920000000

Modified 19 days ago • Private

kenji-shima.nowcast-now-20240620195000

Modified 19 days ago • Private

kenji-shima.forecast-test-20240620200000

Modified 19 days ago • Private

kenji-shima.nowcast-test-20240620195000

Modified Apr 14 • Private

kenji-shima.e90m-test-20240920000000

Modified Apr 14 • Private

kenji-shima.nowcast-20240620195000

New tileset

Tileset hosting

Tileset activity over past 30 days

Active 89

Inactive 258

Tileset tips

What is a tileset?

A tileset is a collection of raster or vector data broken up into a uniform grid of square tiles at 22 preset zoom levels. [Read more.](#)

How to create tilesets

Click 'New tileset' to upload your data. Mapbox renders [vector tiles](#) from your data so you can create styles from it. You can also use Mapbox default tilesets. Read the [vector tiles docs](#) to find out more.

How to use tilesets

Once your vector tiles are ready, you can add them to a new or existing style. First open your style in the style editor. Next, you can either create a new layer with this tileset as the source, or you can change an existing layer's data source to this tileset.

高解像度降水ナウキャスト
(雨雲)であれば、
"mapbox_nowc"など。

HIGHLY CONFIDENTIAL Mapbox, All Rights Reserved.

STEP1: まずはMapboxにご連絡ください

STEP2: Japan Weather Layerを貴社Mapboxアカウントにて有効化

-- 以降、貴社作業 ---

STEP3: Mapbox Studioにて対象のタイルセットを確認

STEP4: MapsSDKまたは、GLJSを使い、貴社システム (フロントエンド)にて当該タイルセットの参照を実装

STEP5: 続いて、スタイリング(色、透明度など)を実装

-- 以降、オプション(気象データにより必須) ---

Option: (風、波など)風のアニメーションを設定 *MapsSDK/GLJSの機能として提供

addSource() => addLayer()により、タイルセットを地図に重畳 通常の Mapbox タイルセットと基本的に同様のフロー

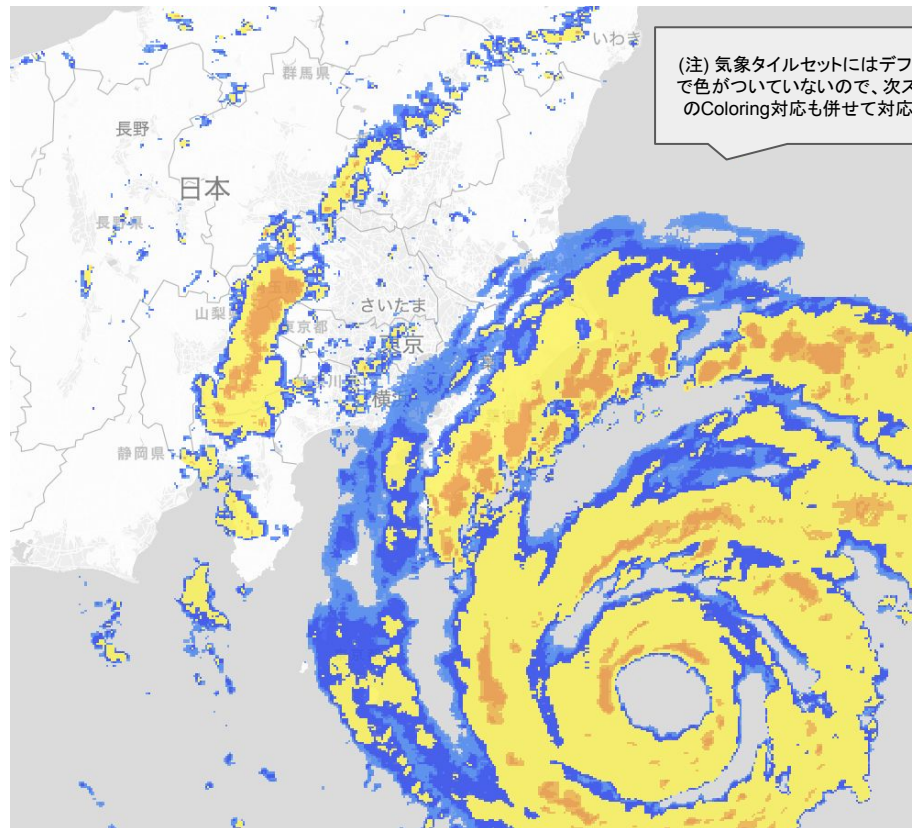
```
map.addSource(  
  'precipitation-source',  
  {  
    type: 'raster-array',  
    url: 'mapbox://mapboxjwl.nowcast'  
  }  
)
```

洪水警報危険度分布(流路)を除き、全て
raster array形式のデータとなるので
"raster-array"の指定が必要

タイルセットIDを指定。常に最新のものが取得できる。(過去分
はタイムスタンプをタイルセット IDにつけて指定。例
:mapboxjwl.nowcast_20250521005045

```
map.addLayer({  
  id: 'precipitation-layer',  
  type: 'raster',  
  source: 'precipitation-source',  
  paint: {  
    'raster-color-range': [0, 200],  
    'raster-color': [  
      'step',  
      ['raster-value'],  
      ...vectorScale()  
    ],  
    'raster-resampling': 'nearest',  
    'raster-color-range-transition': { duration: 0 },  
    'raster-opacity': 0.8  
  },  
})
```

ここではデータの表現の仕方
(スムージングをかけるか、気
象庁のデータに忠実に描画
するか)を選べる。気象庁の
データに忠実に描画する場合
(気象庁のデータに忠実でない
と、気象庁から公開を止め
られるリスクがあるため、忠
実な描画を推奨)、“step”及
び、“nearest”を指定する。



STEP1: まずはMapboxにご連絡ください

STEP2: Japan Weather Layerを貴社Mapboxアカウントにて有効化

-- 以降、貴社作業 ---

STEP3: Mapbox Studioにて対象のタイルセットを確認

STEP4: MapsSDKまたは、GLJSを使い、貴社システム(フロントエンド)にて当該タイルセットの参照を実装

STEP5: 続いて、スタイリング(色、透明度など)を実装

-- 以降、オプション(気象データにより必須) ---

Option: (風、波など)風のアニメーションを設定 *MapsSDK/GLJSの機能として提供

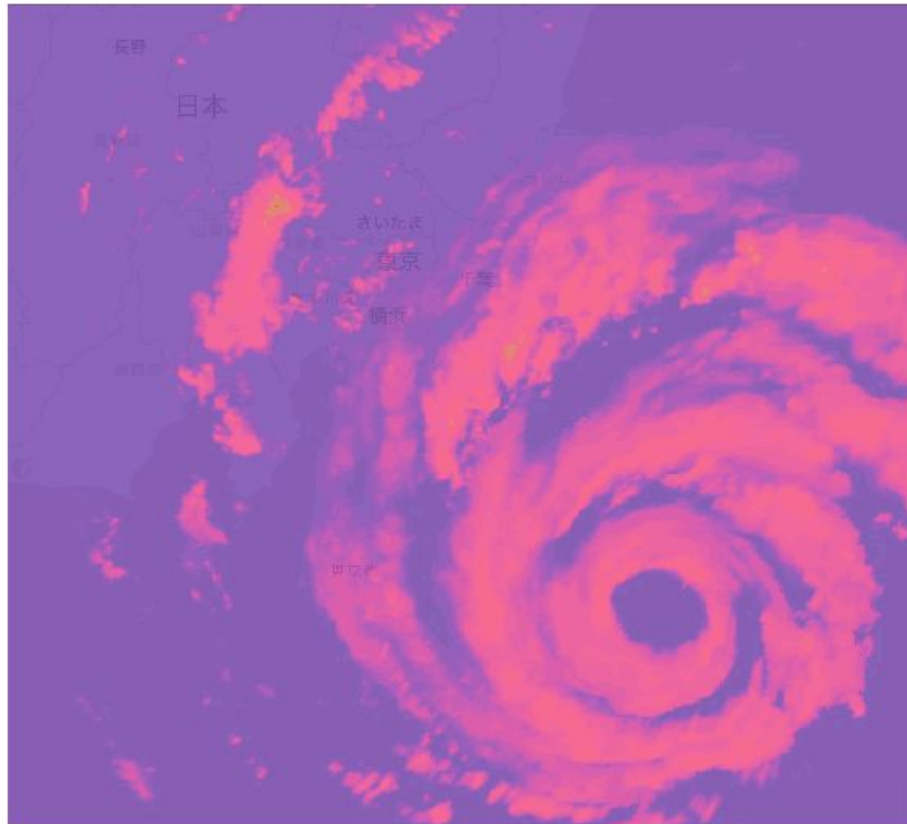
色タイプごとに
これを実装

```
const vectorScale = () => {  
  const domain = [0, 1, 4, 8, 12, 16, 24, 32, 40, 48, 56,  
64, 80, 200];  
  const range = [  
    "rgba(204, 255, 255, 0.0)",  
    "rgba(102, 255, 255, 0.0)",  
    "rgba(0, 204, 255, 0.0)",  
    "rgba(0, 153, 255, 0.8)",  
    "rgba(51, 102, 255, 0.8)",  
    "rgba(51, 255, 0, 0.8)",  
    "rgba(51, 204, 0, 0.8)",  
    "rgba(25, 153, 0, 0.8)",  
    "rgba(255, 255, 0, 0.8)",  
    "rgba(255, 204, 0, 0.8)",  
    "rgba(255, 153, 0, 0.8)",  
    "rgba(255, 80, 102, 0.8)",  
    "rgba(255, 0, 0, 0.8)",  
    "rgba(183, 0, 16, 0.8)"  
  ];  
};
```

降水量

上の配列のn番
目の値が、下の
配列のn番目の
値に対応

色味と透明度
例えば、(138,0,15,0.8)
ならRGB=> 138,0,15
で、不透明度が
0.8(80%)



STEP1: まずはMapboxにご連絡ください

STEP2: Japan Weather Layerを貴社Mapboxアカウントにて有効化

-- 以降、貴社作業 ---

STEP3: Mapbox Studioにて対象のタイルセットを確認

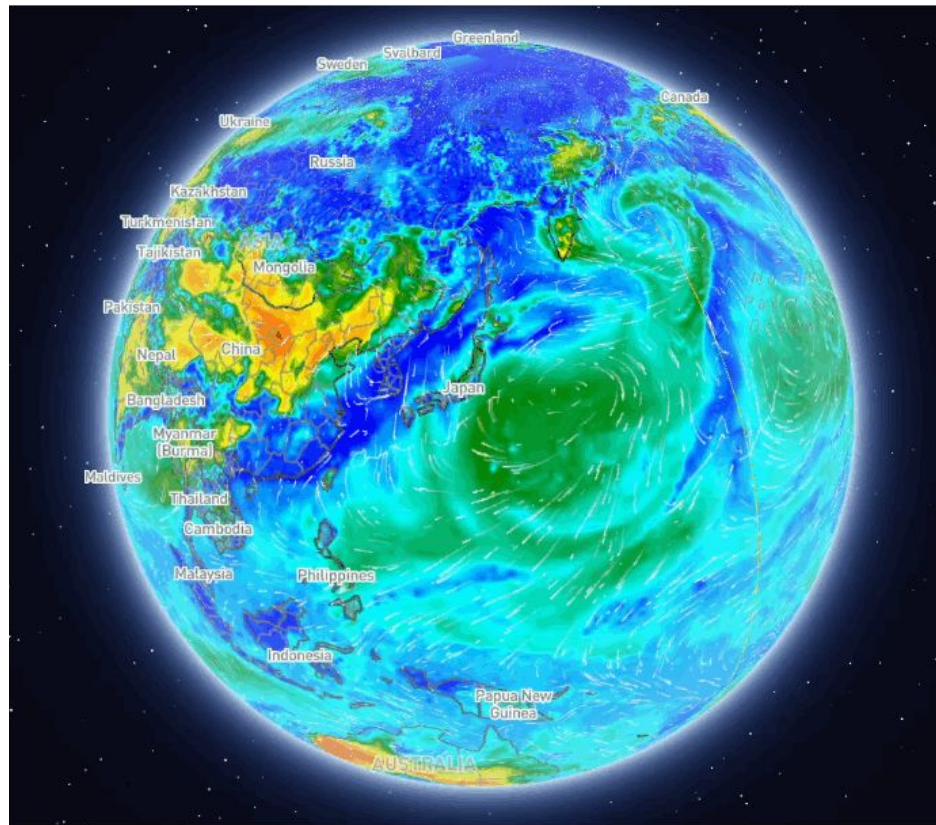
STEP4: MapsSDKまたは、GLJSを使い、貴社システム(フロントエンド)にて当該タイルセットの参照を実装

STEP5: 続いて、スタイリング(色、透明度など)を実装

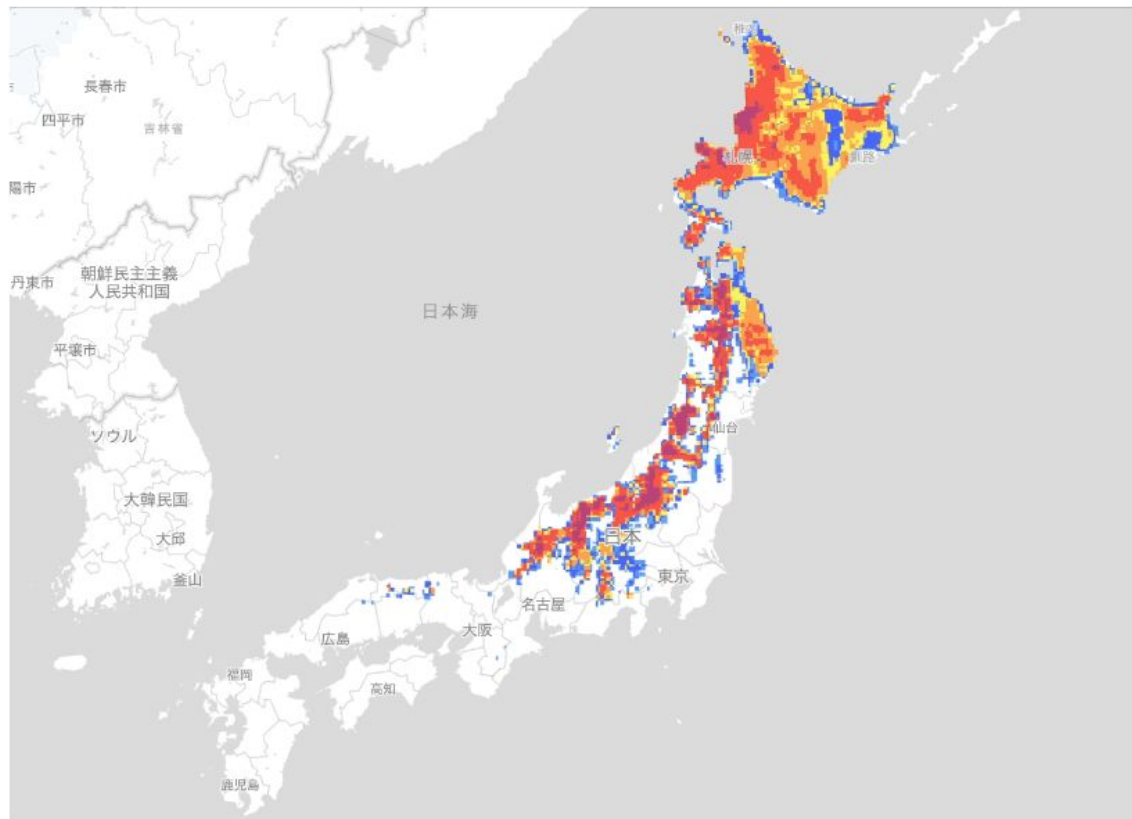
-- 以降、オプション(気象データにより必須) ---

Option:(風、波など)風のアニメーションを設定 *MapsSDK/GLJSの機能として提供

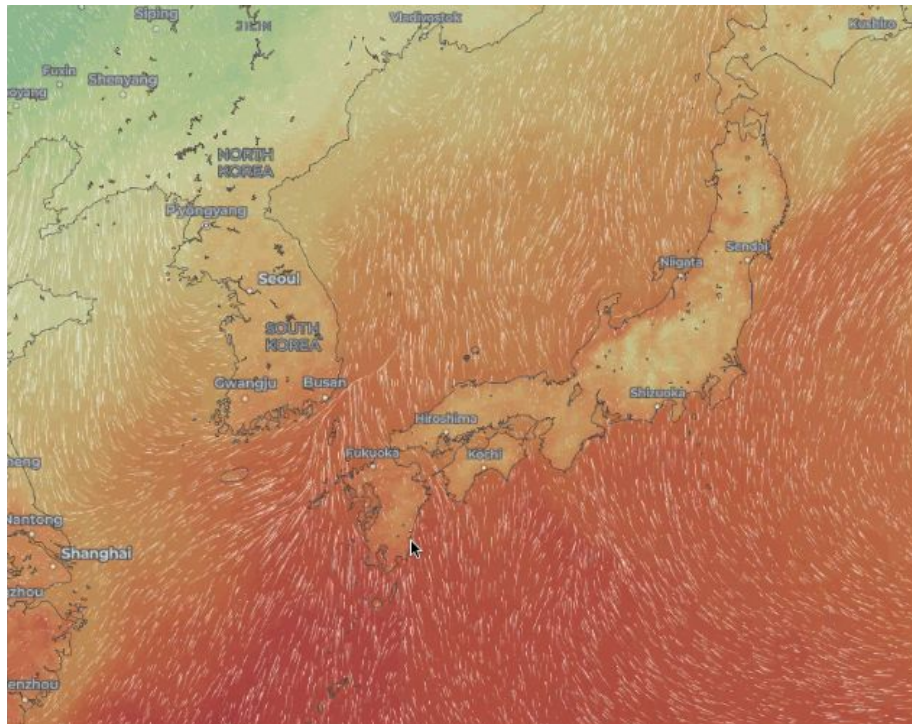

```
map.addSource(  
  'wind-source',  
  {  
    type: 'raster-array',  
    url: 'mapbox://mapboxJWL.wind'  
  }  
)  
  
map.addLayer(  
  {  
    id: 'wind-layer',  
    type: 'raster-particle',  
    source: 'wind-source',  
    layout: {},  
    paint: {  
      'raster-particle-count': 1000, パーティクルの数  
      'raster-particle-max-speed': 10, パーティクルの最大速度  
      'raster-particle-fade-opacity-factor': 0.90, パーティクル  
      がフェードする際の透明度  
      'raster-particle-speed-factor': 0.08, パーティクルの速度  
      'raster-particle-reset-rate-factor': 0.4, パーティクルがラン  
      ダムに再出現する頻度  
      'raster-particle-color': '#fff' パーティクルの色  
    }  
  })
```



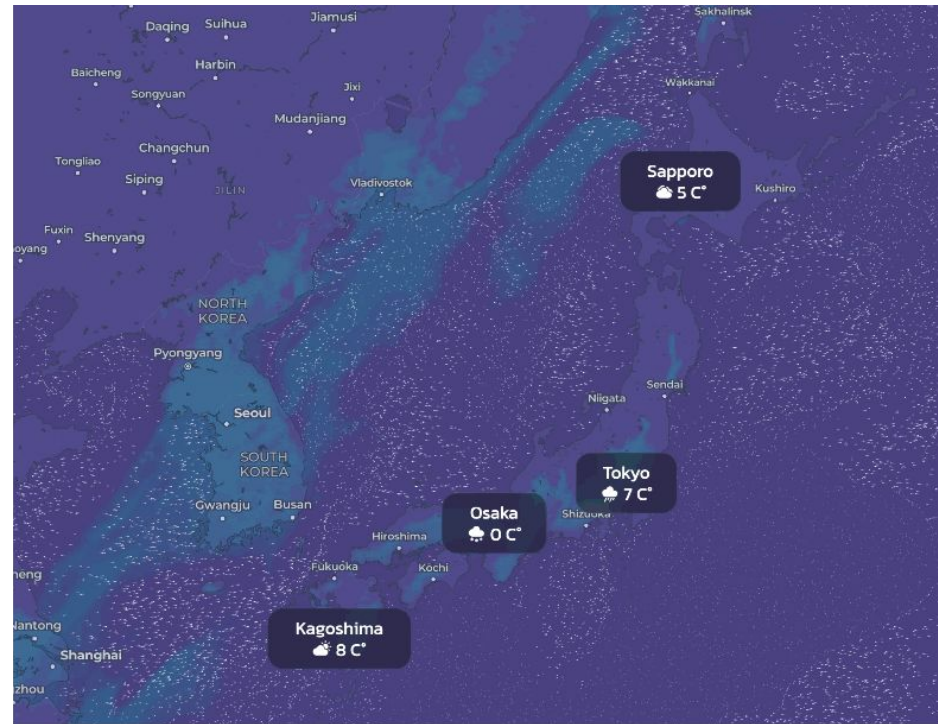
名称	データ提供元	含まれる気象データ
高解像度降水ナウキャスト(1時間先まで)	気象庁	降水量
降水短時間予報(6時間先まで)・降水15時間予報		降水量
降雪短時間予報(6時間先まで)		降雪量
大雨警報(土砂災害)の危険度分布		土砂災害の危険度
洪水警報危険度分布(流路)		洪水の危険度
全球数値予報モデルGPV(全球域)		風速、気圧、気温、相対湿度、露点 (全世界のデータ)
メソ数値予報モデルGPV(MSM)		同上 (日本のみ、上記より高精度)
全球波浪モデルGPV(GWM)		波浪(全世界のデータ)
沿岸波浪モデルGPV(CWM)		同上(日本のみ、上記より高精度)



風速(下地の色味は風の強さを表す)

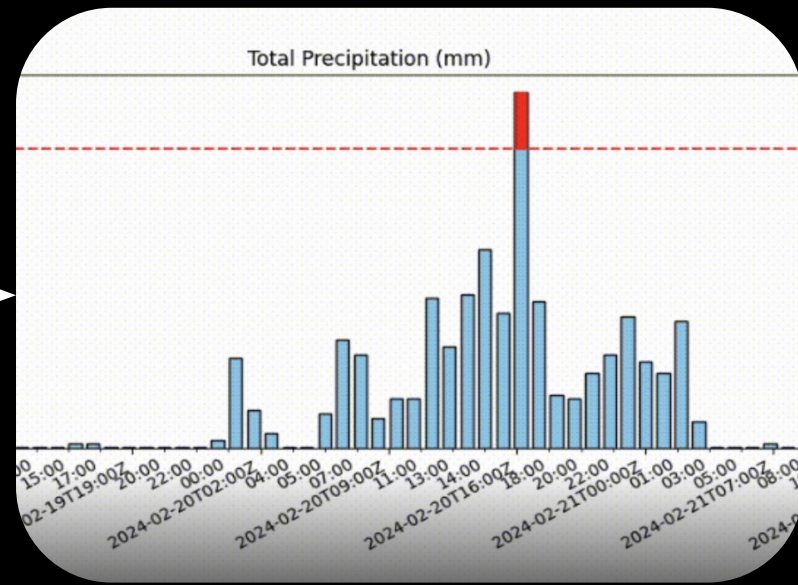


波浪(下地の色味は気温を表す)



Mapbox 3D Weather

Weather x Tilequery API



気象タイルセットから、気象データを取得可能

Tilequery APIをJapan Weather Layerのタイルセットに対して実行することで、
指定緯度経度・時刻の気象データを配信

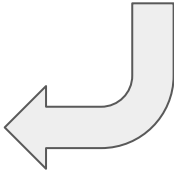
APIコール例(以下で 1コール扱い)

```
https://api.mapbox.com/v4/kenji-shima.nowcast-20240620205000-30m/tilequery/139.6208597052419,35.454866307037264.json?&layers=precipitation&bands=1718916600,1718916900,1718917200,1718917500,1718917800,1718918100,1718918400,1718918700,1718919000,1718919300,1718919600,1718919900,1718920200,1718922600,1718926200,1718929800,1718933400,1718937000,1718938800,1718942400,1718946000,1718949600,1718953200,1718956800,1718960400,1718964000,1718967600&access_token=ここにトークンを入れる
```

リターン

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [16],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718916600",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    },
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [16],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718916900",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    },
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [10],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718917200",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    },
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [9],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718917500",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    },
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [9],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718917800",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    },
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [11],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718918100",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    },
    {
      "type": "Feature",
      "id": null,
      "geometry": {
        "type": "Point",
        "coordinates": [139.6208597052419, 35.454866307037264]
      },
      "properties": {
        "val": [10],
        "tilequery": {
          "layer": "precipitation",
          "band": "1718918400",
          "zoom": 9,
          "units": "W/(m^2)"
        }
      }
    }
  ]
}
```

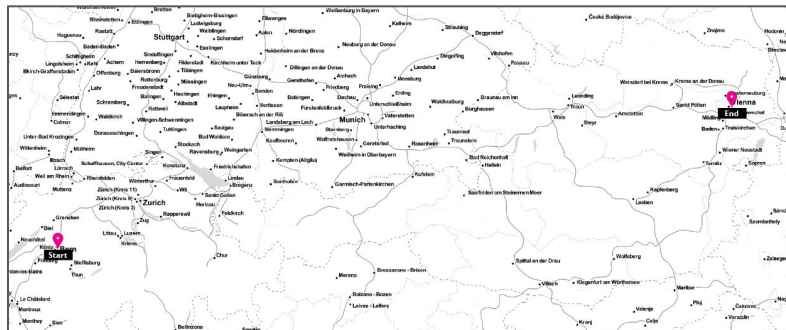
“coordinate” = 緯度経度	“Val” = 5分間降水強度 (ml)	“Band” = 時刻	備考
139.6208597052419, 35.454866307037264 場所はピンポイント、半径両方 (半径だと半径内の全ての値を 取る)で指定可能	16	1718916600	現在時刻 (20240620205000 in UTC)
	16	1718916900	5分後
	10	1718917200	10分後
	9	1718917500	15分後
	9	1718917800	20分後
	11	1718918100	25分後
	10	1718918400	30分後



想定利用シーン

ユースケース: 到達所要時間算出

輸・配送先までの雨雲を検知し、より高精度な ETAの算出が可能

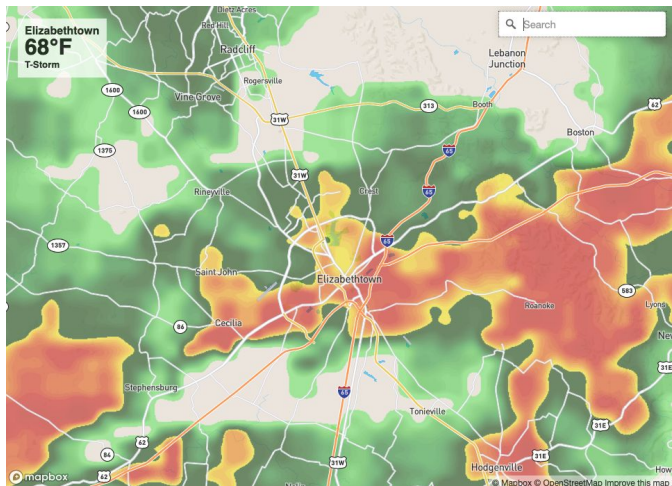


メリット

- Mapbox側で雨雲情報のアップロード・更新を行うのでメンテナンス費用低減
- Tilequery APIと組み合わせることでドライバーが通るルートを基にETA算出を行うことが可能

ユースケース:ダイナミックプライシング

雨が降るエリアを予測し、配送ドライバーの単価算出



メリット

- ドライバーだけでなく、ユーザーの手数料変更にも利用可能。
- ドライバー側Appにも雨雲情報を表示する事で降雨時のドライバー不足を解消。

