

OGC API – Moving Features, an introduction with MF-API Server based on pygeoapi and MobilityDB

Wijae Cho, Taehoon Kim, Kyoungsook Kim

Data Platform Research Team

Artificial Intelligence Research Center

National Institute of Advanced Industrial Science and Technology

(AIST)



Seoul Hall of Urbanism & Architecture. 28th Nov.~2nd Dec.

Outline

1. Introduction of STINUUM
2. Installation
3. Practice with offline STINUUM
4. Practice with online STINUUM

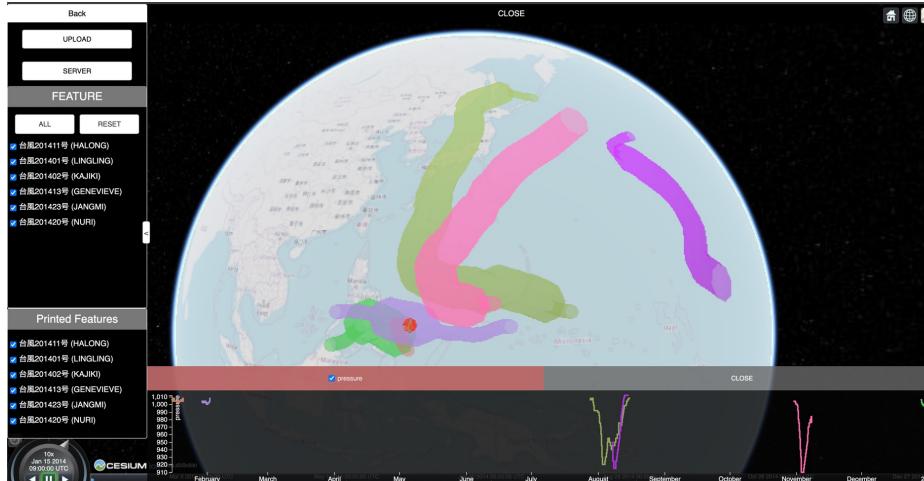
Introduction of STINUUM

1. What is STINUUM?
2. System structure of STINUUM
3. Components of STINUUM
4. How to visualize MF-JSON data in the STINUUM
 1. Visualize TemporalGeometry object
 2. Visualize TemporalProperties object
 1. Measure, Text types
 2. Image type

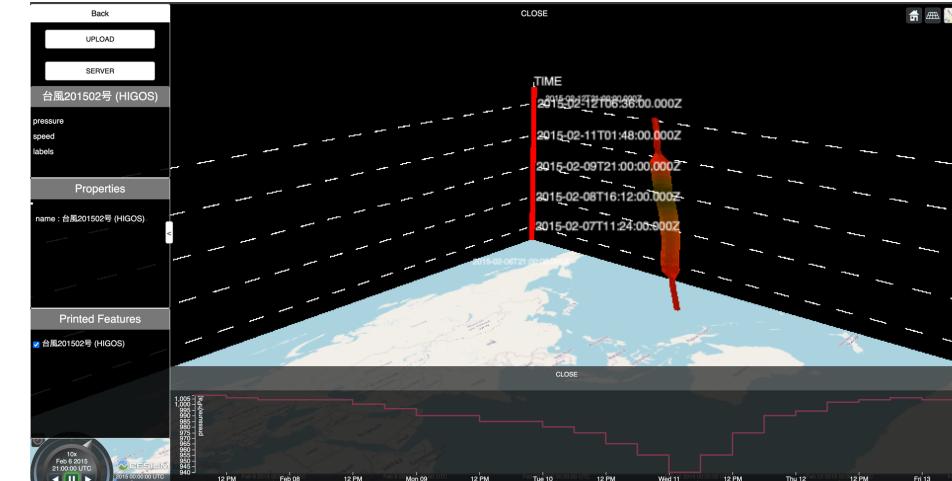
Introduction of STINUUM

- **What is STINUUM (Spatio-Temporal continua on Cesium)**

- https://github.com/aistairc/mf-cesium/tree/mf-cesium_api
- STINUUM is a JavaScript library to visualize and analyze moving objects on Cesium
- STINUUM renders OGC Moving Features JSON (MF-JSON) data to explore the trajectories of moving objects across space and time
- The main characteristics of STINUUM are as follows:
 - Diverse geometry types to represent movements
 - Multiscale data analysis in space and time
 - Highly accessible and lightweight deployment



Visualize the Moving Feature

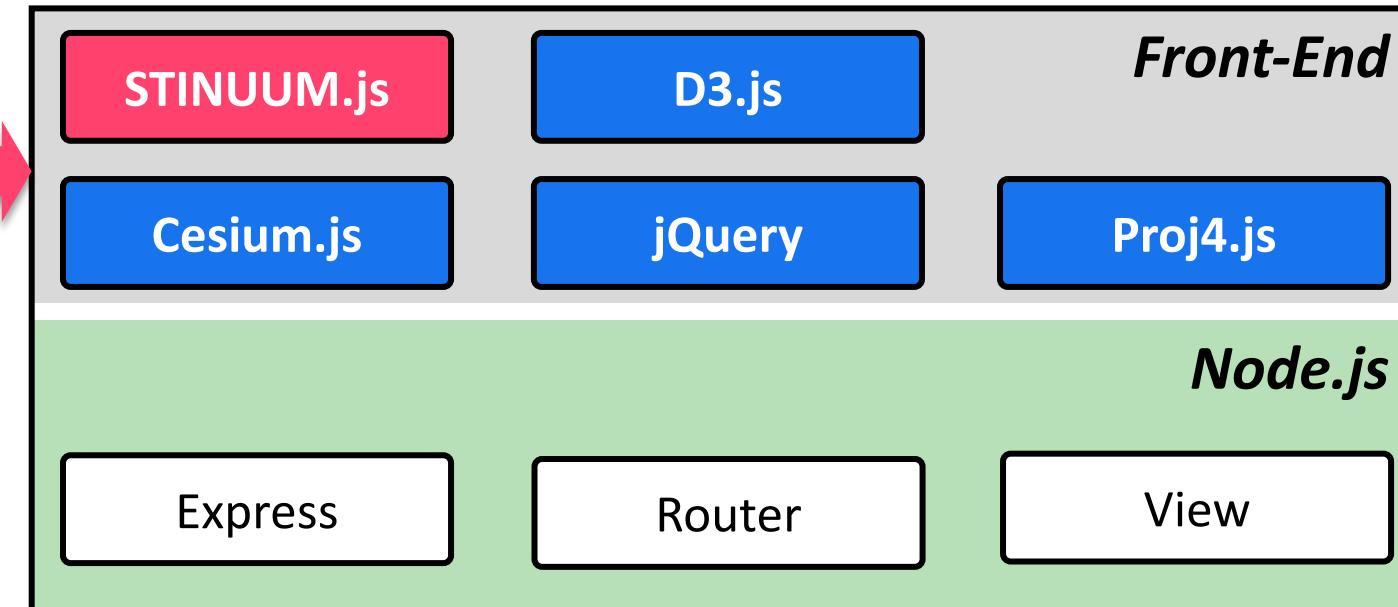
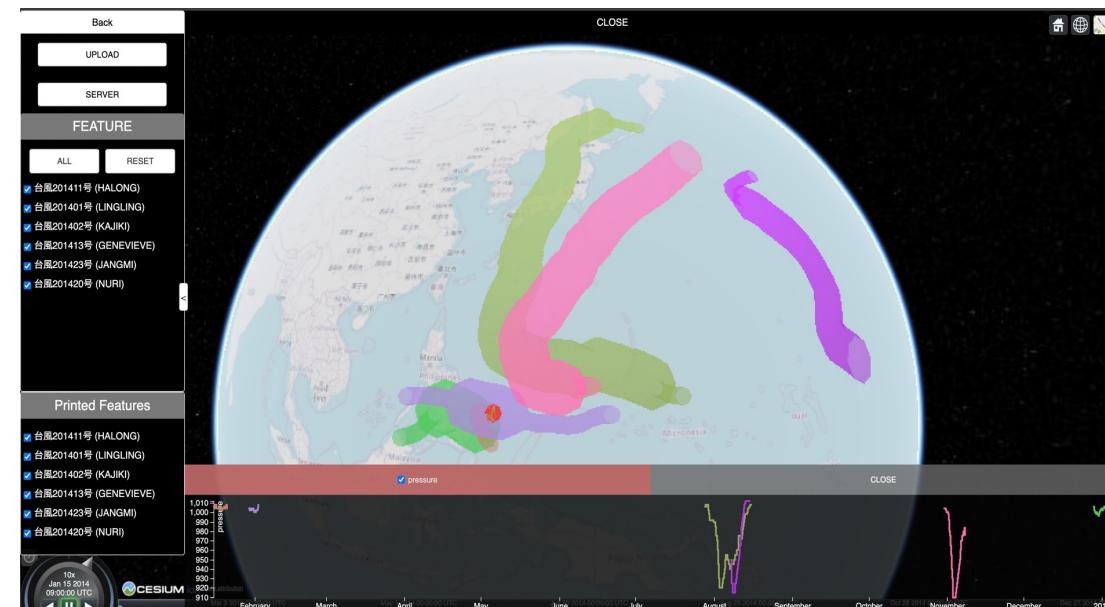


Visualize the Moving Feature in Space-time map

Introduction of STINUUM

- **System structure of STINNUM**

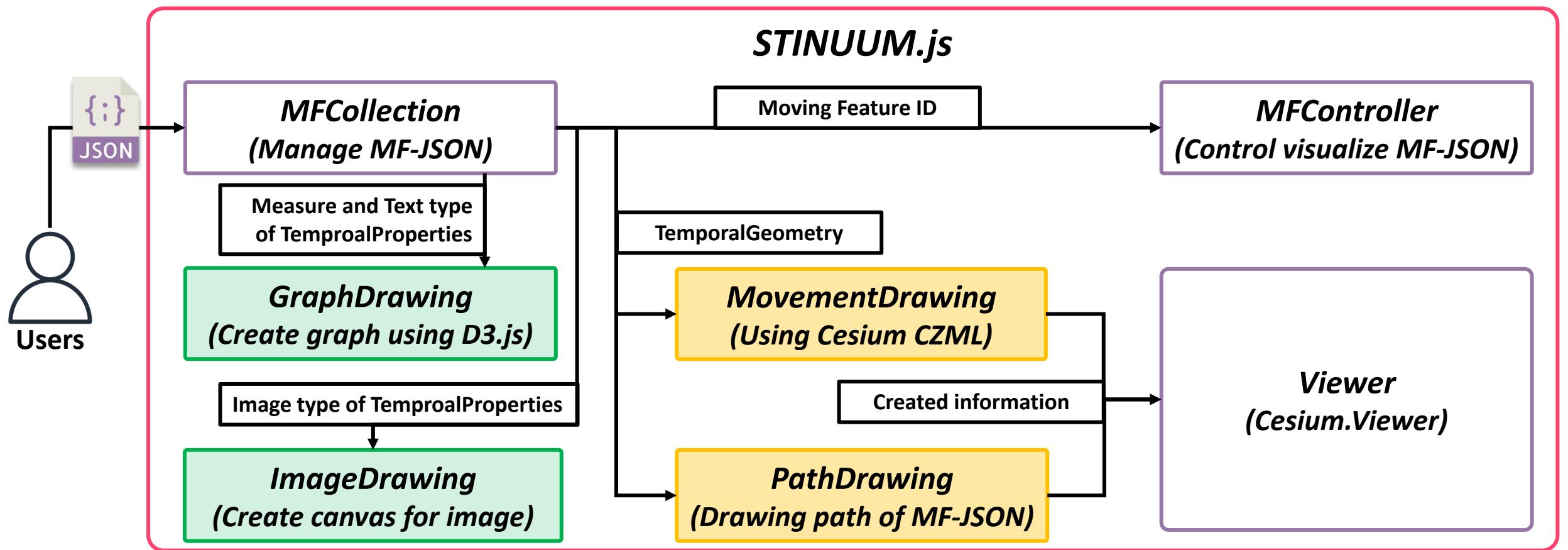
- **STINUUM.js**: Provides functionality for visualizing and controlling moving features
- **Cesium.js**: Provides animation control and visualization tools for visualizing moving features
- **D3.js**: Creates graphs using *TemporalProperties* object
- **Proj4.js**: Used for coordinate transformations in *TempoproalGemetry* object
- **jQuery**: Used for UI creation and controls
- **Node.js**: Used to run and deploy web applications using **STINUUM.js**



Introduction of STINUUM

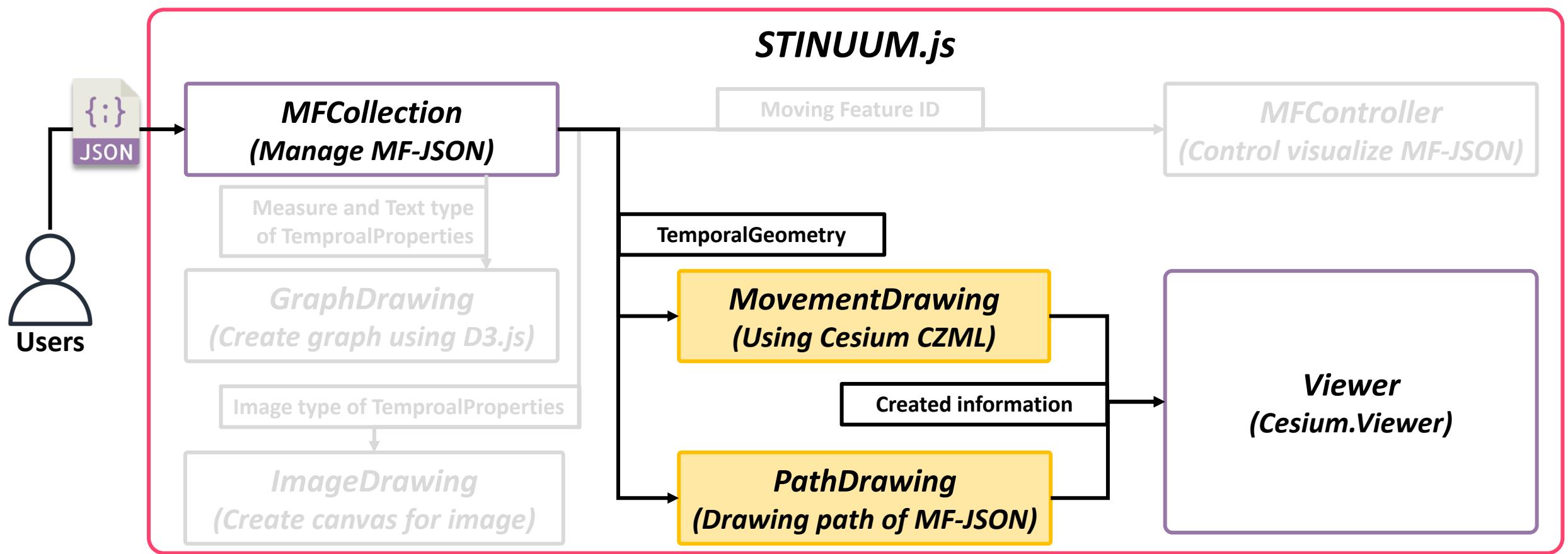
- Components of STINNUM

- STINUUM.js provides fundamental methods to handle MF-JSON data



Introduction of STINUUM

- How to visualize MF-JSON data in the STINUUM
 - Visualize *TemporalGeometry* object



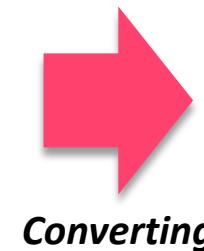
Introduction of STINUUM

- How to visualize MF-JSON data in the STINUUM

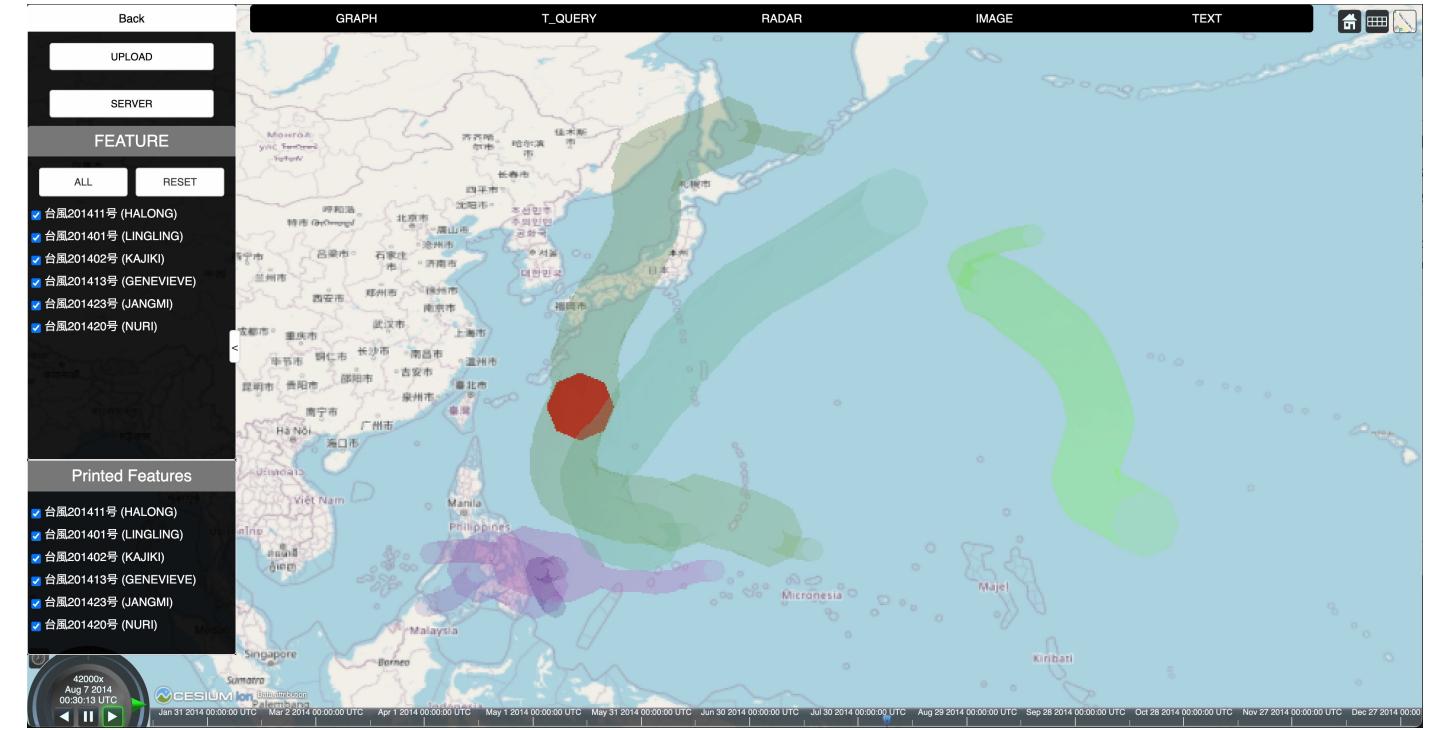
- Visualize *TemporalGeometry* object

- Converting the *TemporalGeometry* information to CZML format and visualizing it
 - CZML is a data format used to represent moving objects in Cesium
 - Easily visualize moving objects in Cesium without complex implementation using APIs

```
"temporalGeometry": {
  "type": "MovingPolygon",
  "datetimes": [
    "2011-07-14 22:01:01",
    "2011-07-14 22:01:12",
    ...
  ],
  "coordinates": [
    [ 139.757083, 35.627701, 0.5 ],
    [ 139.757399, 35.627701, 2.0 ],
    [ 139.757555, 35.627688, 4.0 ],
    [ 139.757651, 35.627596, 4.0 ],
    [ 139.757083, 35.627701, 0.5 ] ],
    ...
  ],
}
```



Converting

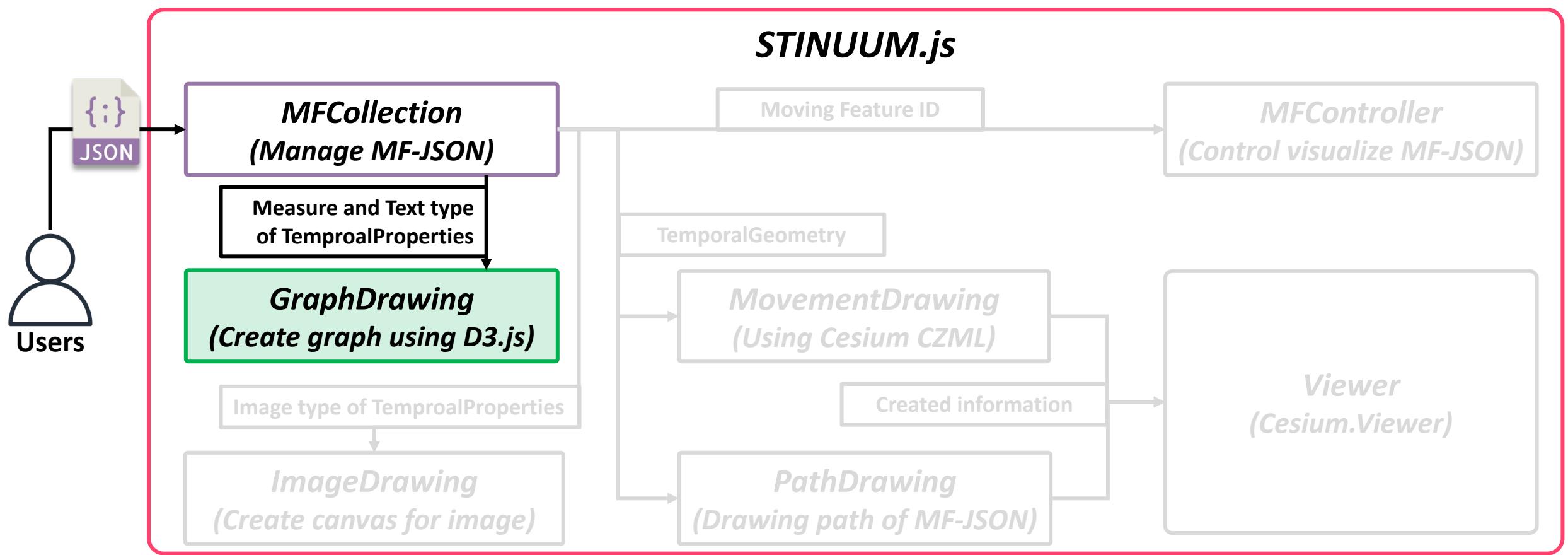


TemporalGeometry of Moving Feature

Visualize temporal geometry (MovingPolygon) in Cesium

Introduction of STINUUM

- How to visualize MF-JSON data in the STINUUM
 - Visualize “Measure” and “Text” types of *TemporalProperties*



Introduction of STINUUM

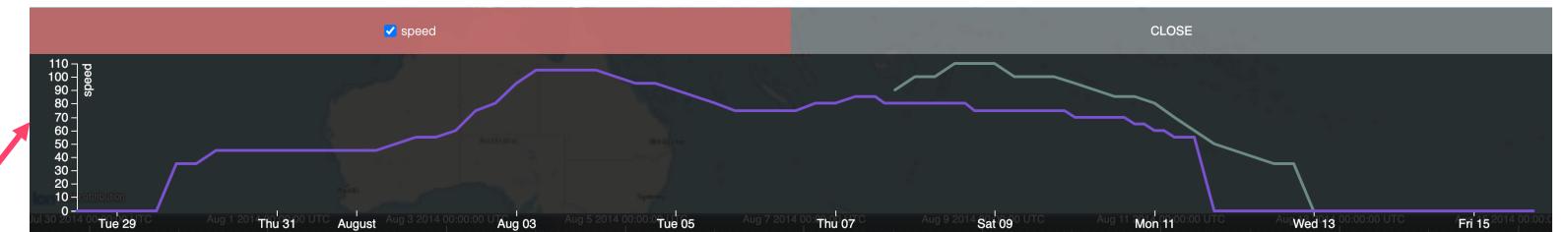
- How to visualize MF-JSON data in the STINUUM

- Visualize *TemporalProperties* object

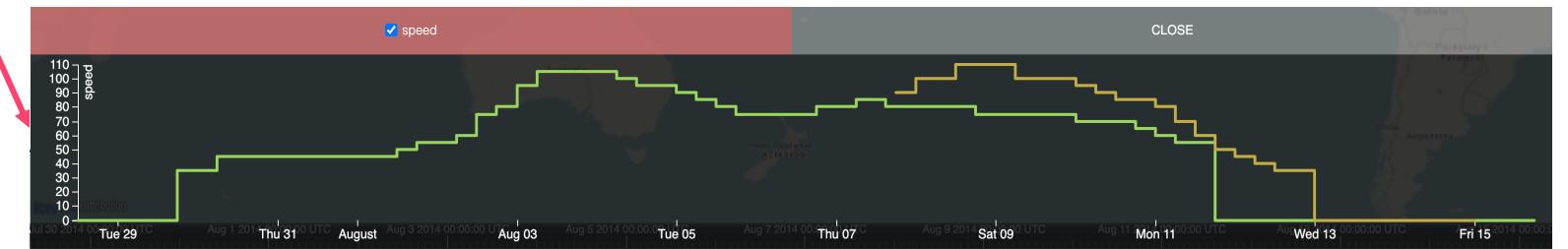
- Using the **D3.js** to create a graph for the “Measure” and “Text” types of temporal property
 - x-axis: time information in the selected temporal property
 - y-axis: value information in the selected temporal property
 - The interpolation of each temporal property is displayed using the functions supported by **D3.js**
 - Discrete, Step, Linear, Regression

```
"temporalProperties": [
  {
    "datetimes": [
      "2011-07-14 22:01:01",
      "2011-07-14 22:01:12",
      ...
    ],
    "speed": {
      "type": "Measure",
      "form": "",
      "values": [ 100, 200, 300, ... ],
      "interpolation": "Linear"
    }
  },
  ...
]
```

TemporalProperties of Moving Feature



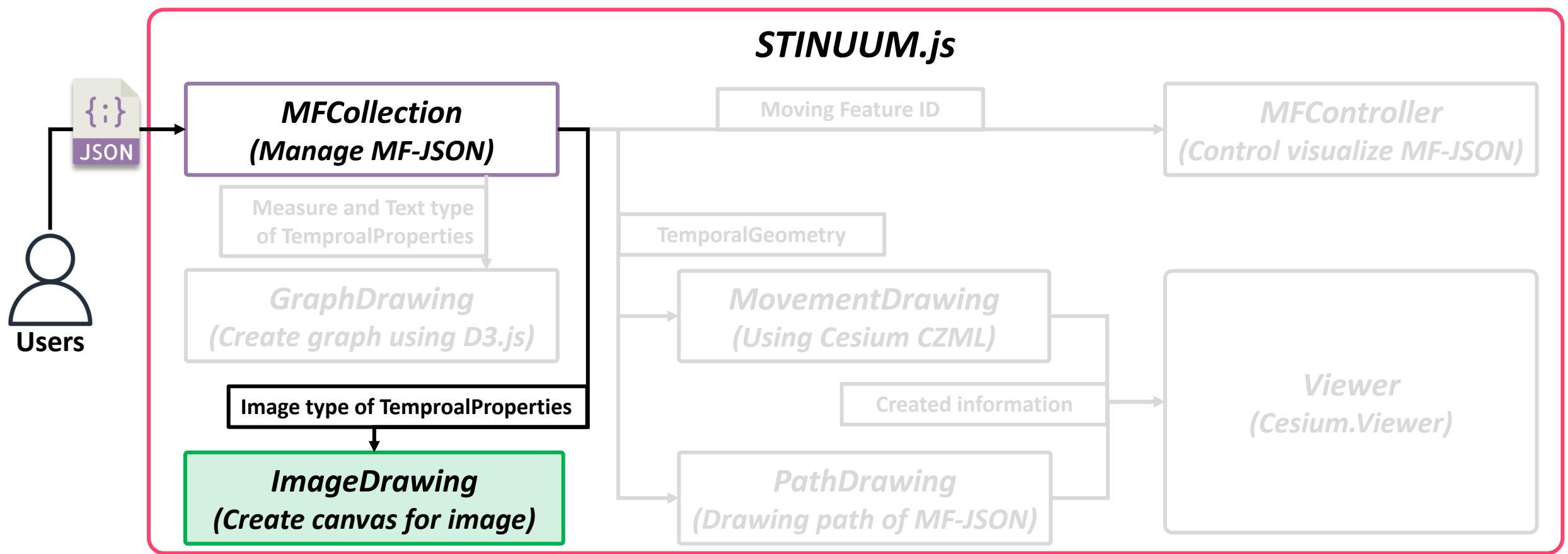
Linear graph of speed



Step graph of speed

Introduction of STINUUM

- How to visualize MF-JSON data in the STINUUM
 - Visualize “Image” type of *TemporalProperties*



Introduction of STINUUM

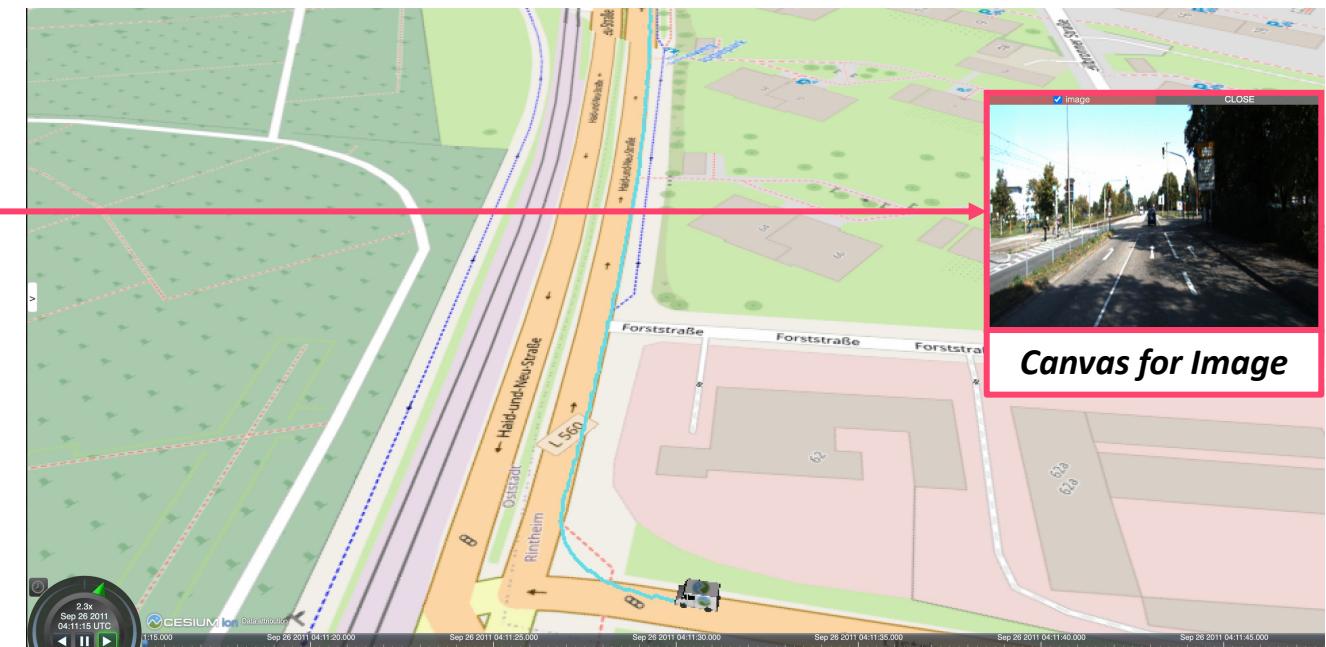
- How to visualize MF-JSON data in the STINUUM

- Visualize *TemporalProperties* object

- Using the HTML Canvas to display the “Image” type of temporal property
 - Visualize images that match the date-time in the selected temporal property using **Cesium.EventHandler**

```
"temporalProperties": {
  "datetimes": [
    "2011-07-14 22:01:01",
    "2011-07-14 22:01:12",
    ...
  ],
  "image": {
    "type": "Image",
    "values": [
      "VBORw0KGgoAAAANU.....",
      "VBORw0KGgoAAAANU.....",
      ...
    ],
    "interpolation": "Step"
  }
}, ...]
```

TemporalProperties of Moving Feature



Display the image with Moving Feature

Installation

1. Install STINUUM
2. How to connect to the MF-API server
3. MF-API-Handler

Installation

• Install STINUUM

- URL: https://github.com/aistairc/mf-cesium/tree/mf-cesium_api
- git clone <https://github.com/aistairc/mf-cesium>
 - git fetch origin → git checkout mf-cesium_api
- Install nvm(Please refer to the README)
 - Windows
 - Download the nvm-setup.zip in <https://github.com/coreybutler/nvm-windows/releases>
 - » <https://github.com/coreybutler/nvm-windows/wiki>
 - Unzip nvm-setup.zip and run nvm-setup.exe
 - MacOS (using terminal)
 - brew install nvm
 - Open the .bash_profile or .zshrc
 - » vi ~/.bash_profile
 - Add the environment value of nvm
 - »

```
export NVM_DIR="$HOME/.nvm" [ -s "/usr/local/opt/nvm/nvm.sh" ] && . "/usr/local/opt/nvm/nvm.sh" # This
loads nvm [ -s "/usr/local/opt/nvm/etc/bash_completion.d/nvm" ] && .
"/usr/local/opt/nvm/etc/bash_completion.d/nvm" # This loads nvm bash_completion
```

Installation

• Install STINUUM

- Install nvm
 - Ubuntu (using terminal)
 - sudo apt install curl
 - curl https://raw.githubusercontent.com/creationix/nvm/master/install.sh | bash
 - source ~/.bashrc
 - Check installed nvm with terminal
 - nvm -v
 - Install node.js
 - nvm install 20.10.0
 - nvm use 20.10.0
 - node -v
- Install STINUUM
 - 1) Cd /path_to/mf-cesium/Stinuum Web
 - 2) npm install
- Start STINUUM
 - node app.js

Installation

- **How to connect to the MF-API Server**

- docker pull ghcr.io/taehoonk/mf-api-server:1.1
- docker run -p 8085:8085 -p 25432:5432 -d --name mf-api-server ghcr.io/taehoonk/mf-api-server:1.1
 - Port number to access*
- docker exec mf-api-server ./run.sh
- Connect to the MF-API Server
 - **localhost:8085**

Installation

- **MF-API Handler**

- URL: https://github.com/aistairc/mf-cesium/tree/mf-cesium_api/MFAPIHandler
 - cd /path_to/mf_cesium/MFAPIHandler
 - pip install –r requirements.txt
- Simple program to register the Moving Feature data to MF-API Server
 - Register new Moving Feature Collection
 - Uploading the Moving Feature data

Practices with offline STINUUM

1. Notice about sample data location (MF-JSON data)
2. MF-JSON data upload
3. Visualize the temporal geometry of the moving feature
4. Animate the visualized moving feature
5. Select sub-temporal geometry (Time query)
6. Change the view modes (static map, global view, space-time cube view)
7. Visualize the temporal property (Measure type)
8. Visualize the Measure type with temporal geometry (highlight)
9. Visualize the temporal property (Text type)
10. Visualize the temporal property (Image type)

Practice with offline STINUUM

- Notice about sample data location (MF-JSON data)

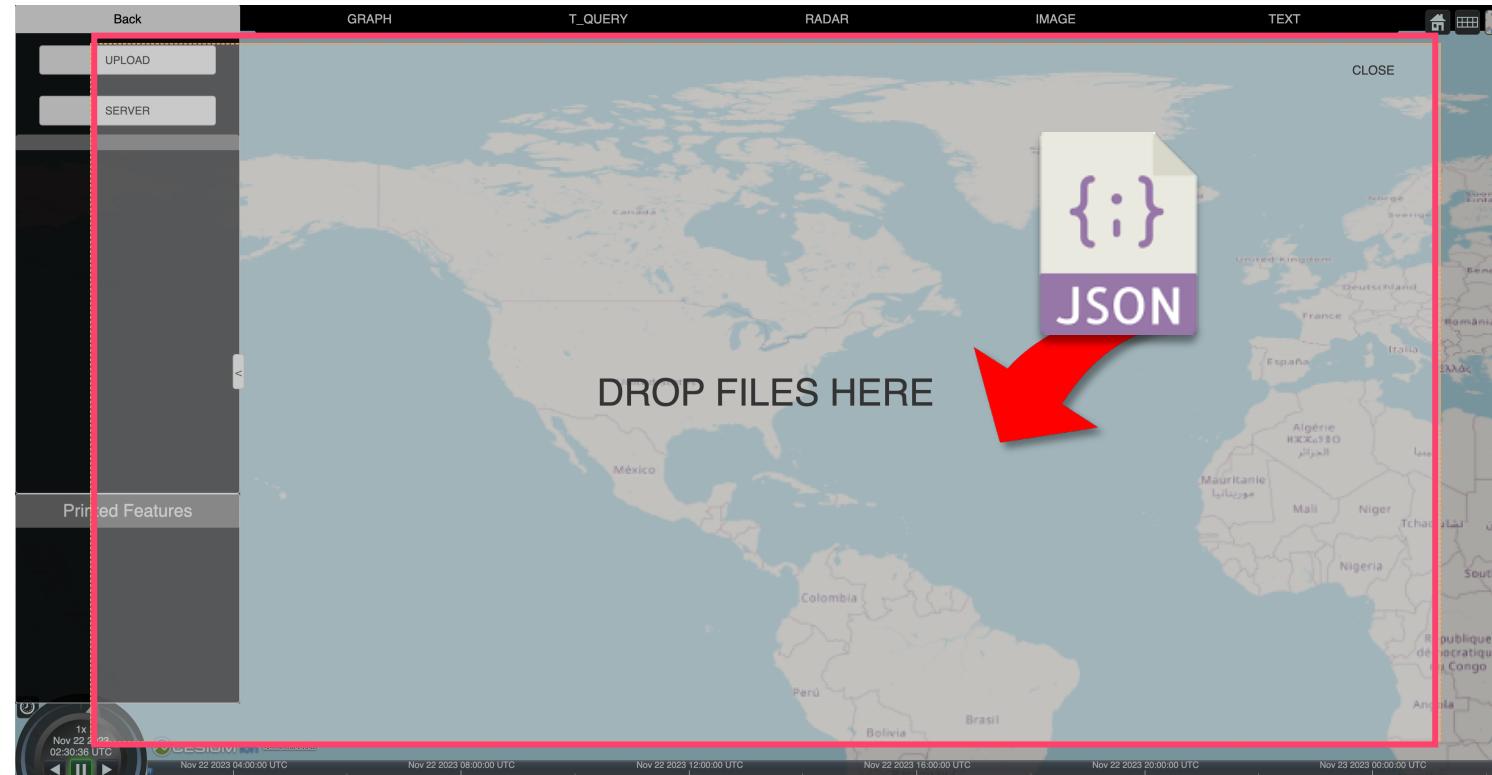
- Path of file: /path_to/mf_ceisum/Stinuum Web/data/MF-JSON Prism/PracticeDataSet
- Sample Moving Feature data
 - **mfjson_point_car_with_image.json & mfjson_point_for_mf_api_server.json**
 - MF-JSON temporalGeometry type: MovingPoint
 - Homepage: <https://www.cvlibs.net/datasets/kitti/index.php>
 - Vehicle movement information with image in German
 - » Create MF-JSON using data
 - KITTI data
 - **mfjson_polygon_multiple_typhoon.json**
 - MF-JSON temporalGeometry type: MovingPolygon
 - Data of typhoon movement
 - **mfjson_point_from_mobilitytwin.json**
 - MF-JSON temporalGeometry type: MovingPoint
 - Homepage: <https://mobilitytwin.brussels/>
 - Information on public transportation in Brussel

Practice with offline STINUUM

- **MF-JSON data upload**

- Visualize MF-JSON file by dragging and dropping

- Path of file: **Stinuum Web/data/MF-JSON Prism/PracticeDataSet/mfjson_polygon_multiple_typhoon.json**
 - Visualization using text information of MF-JSON

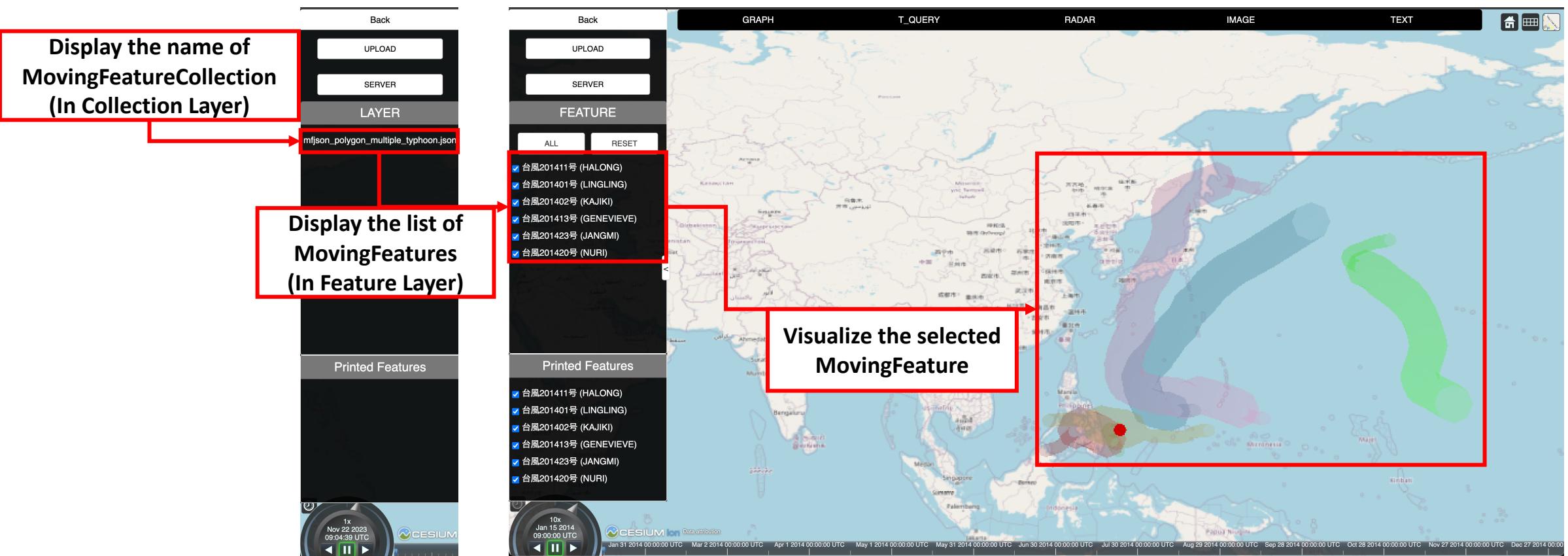


Upload MF-JSON using file

Practice with offline STINUUM

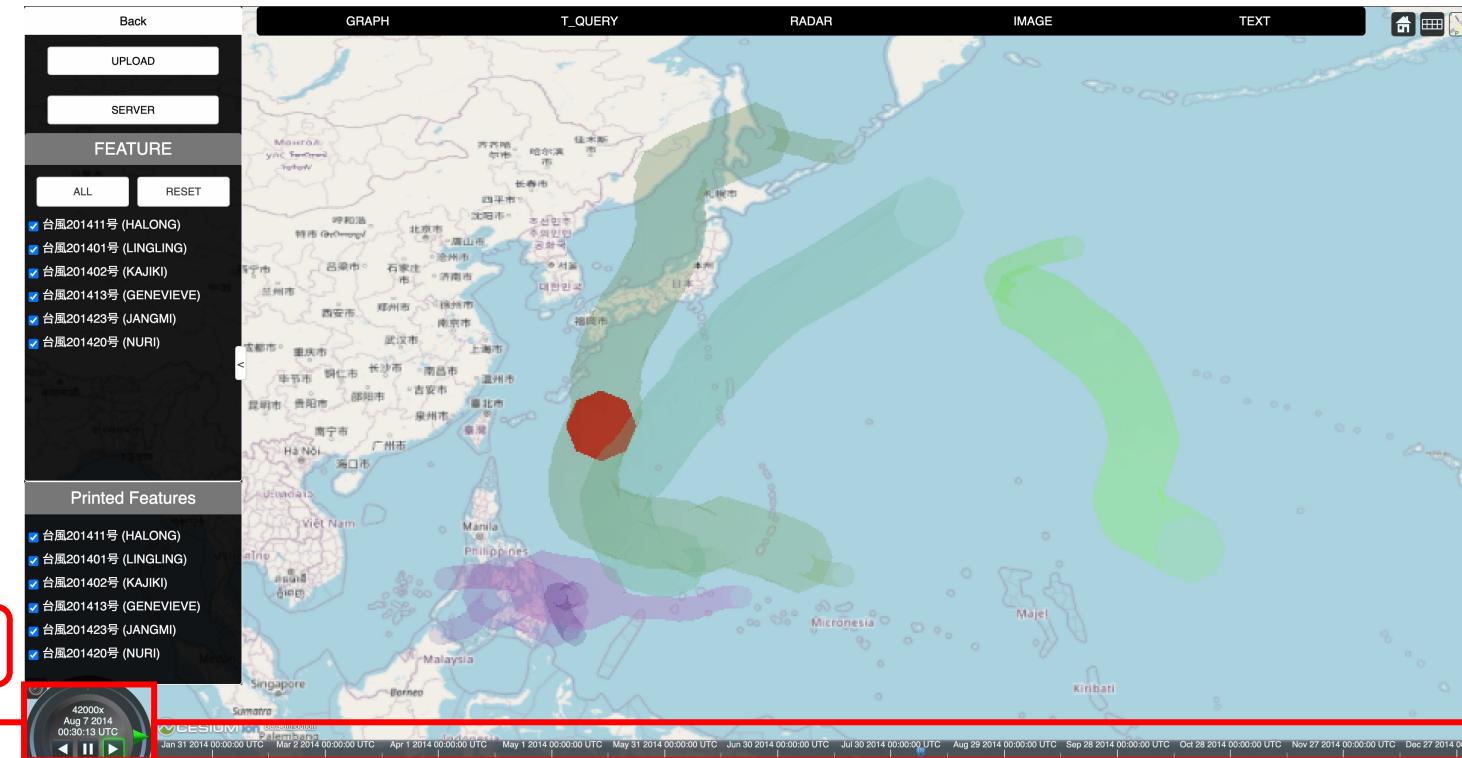
- **Visualize the temporal geometry of the moving features**

- When the MF-JSON upload is complete, you'll see the information in the left menu (Collection Layer)
- Click the uploaded **MovingFeatureCollection** (or **MovingFeature**) object instance
 - A list of **MovingFeatures** will be displayed (Feature Layer)



Practice with offline STINUUM

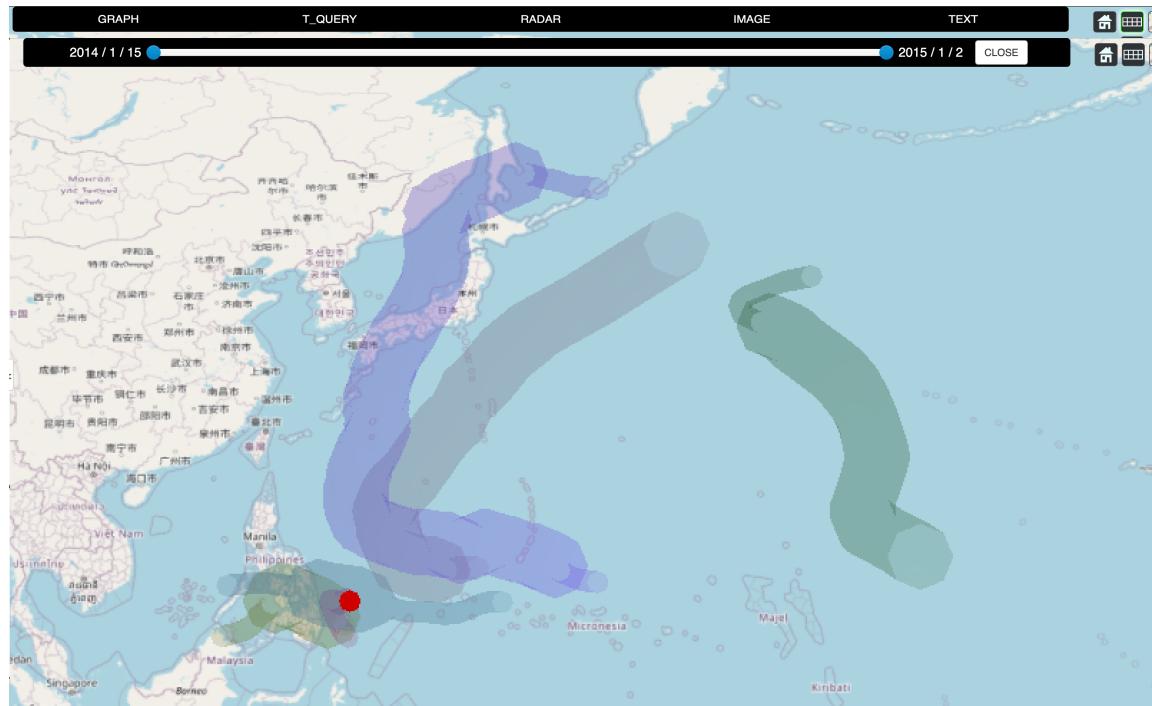
- Animate the visualized moving feature
 - Animating visualized moving features using the animation function provided by Cesium
 - Dynamically visualize the trajectory of the moving features according to time



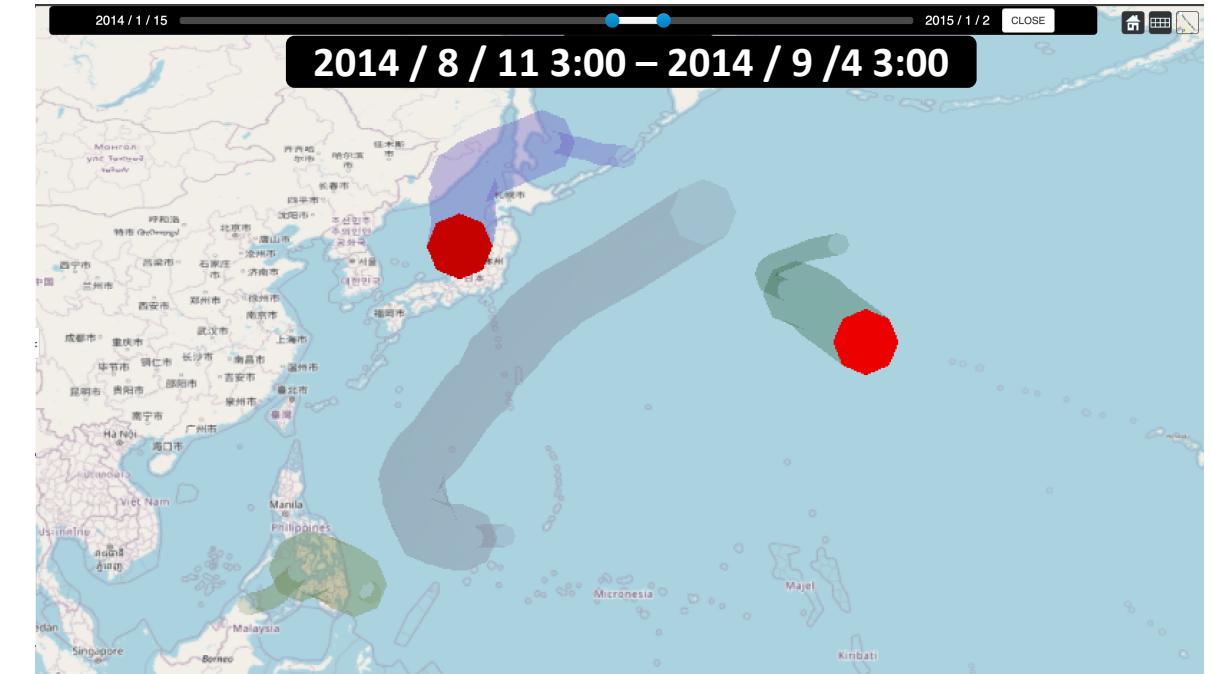
Practice with offline STINUUM

- Select the sub-temporal geometry (time query)

- The time query is used to check the movement of the moving features in a specific time period
 - The temporal property is also displayed in a specific time period



Trajectories display for the entire time period



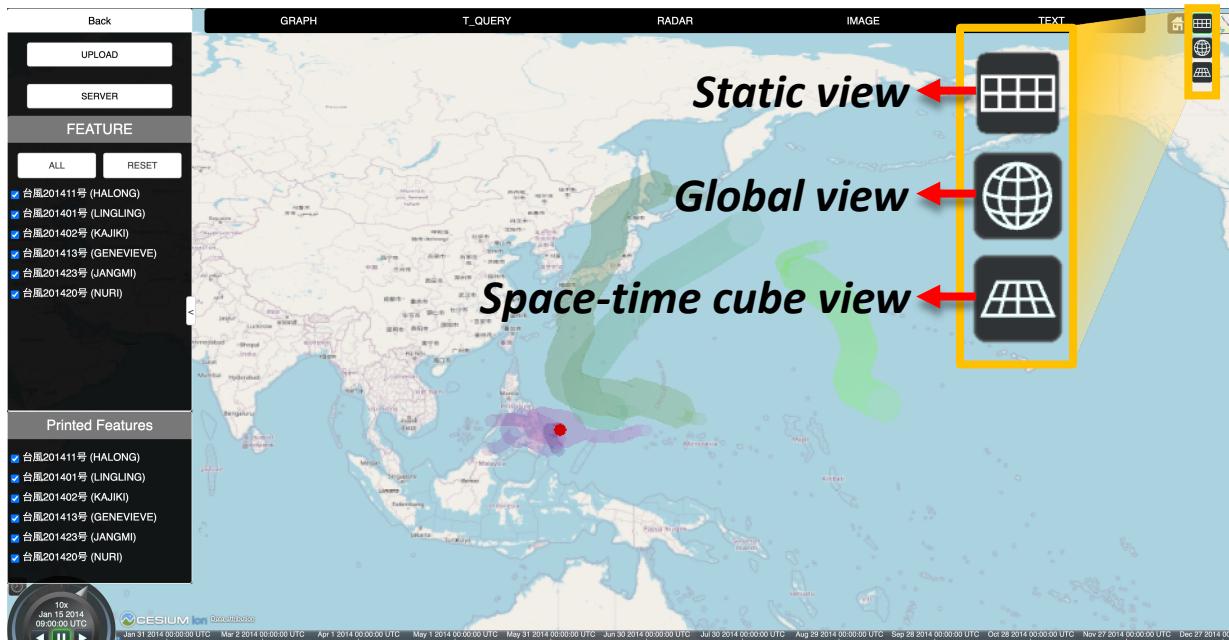
Time query results

Practice with offline STINUUM

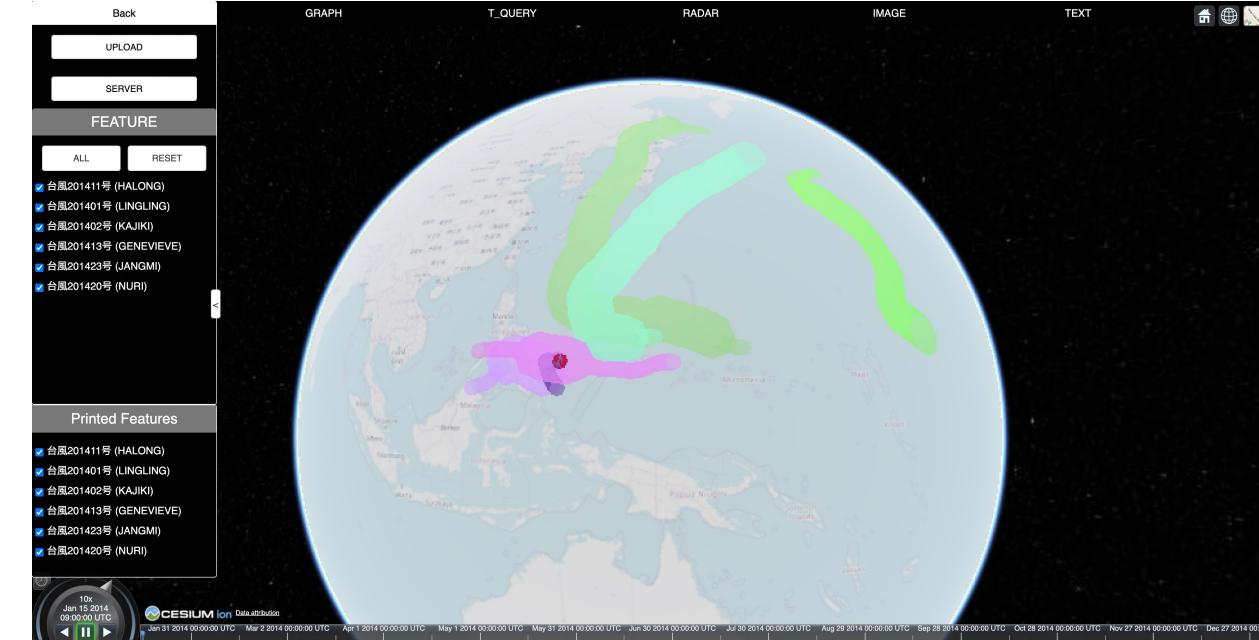
- Change the view modes (Static, Global, and Space-time cube view)

- STINUUM supports three view modes: Static, Global, and Space-time cube view

- Users can select (and change) the view mode using the map toolbox in the top right corner
- **Static view:** (Default) The map is viewed top-down with an orthographic projection
- **Global view:** A traditional 3D perspective view of the globe



Static View (2D Map)



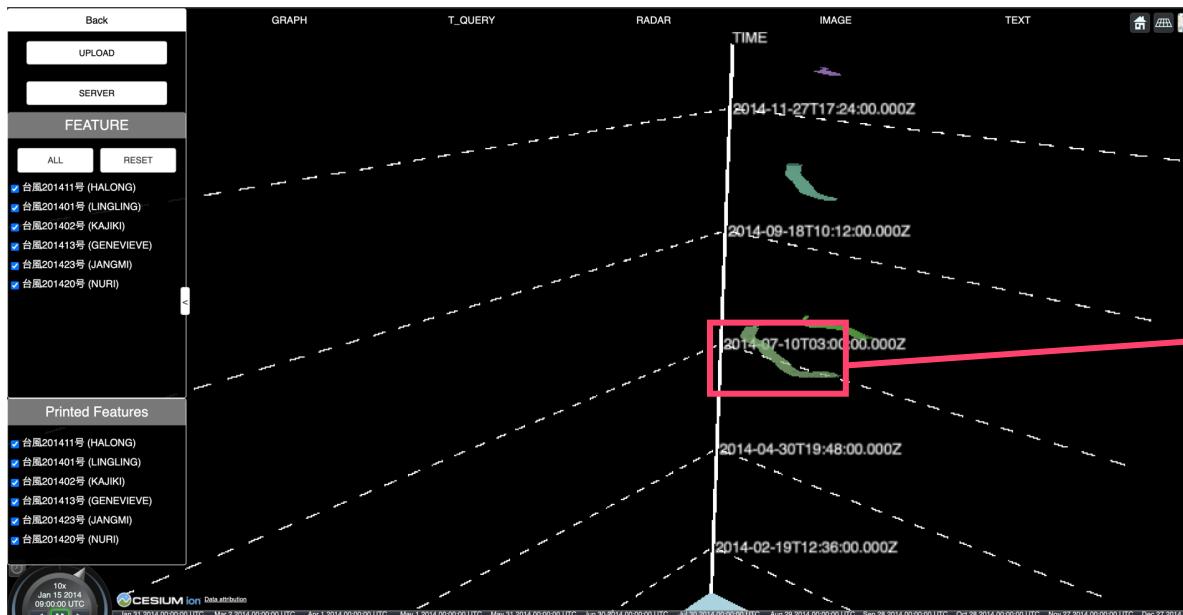
Global View (3D Map)

Practice with offline STINUUM

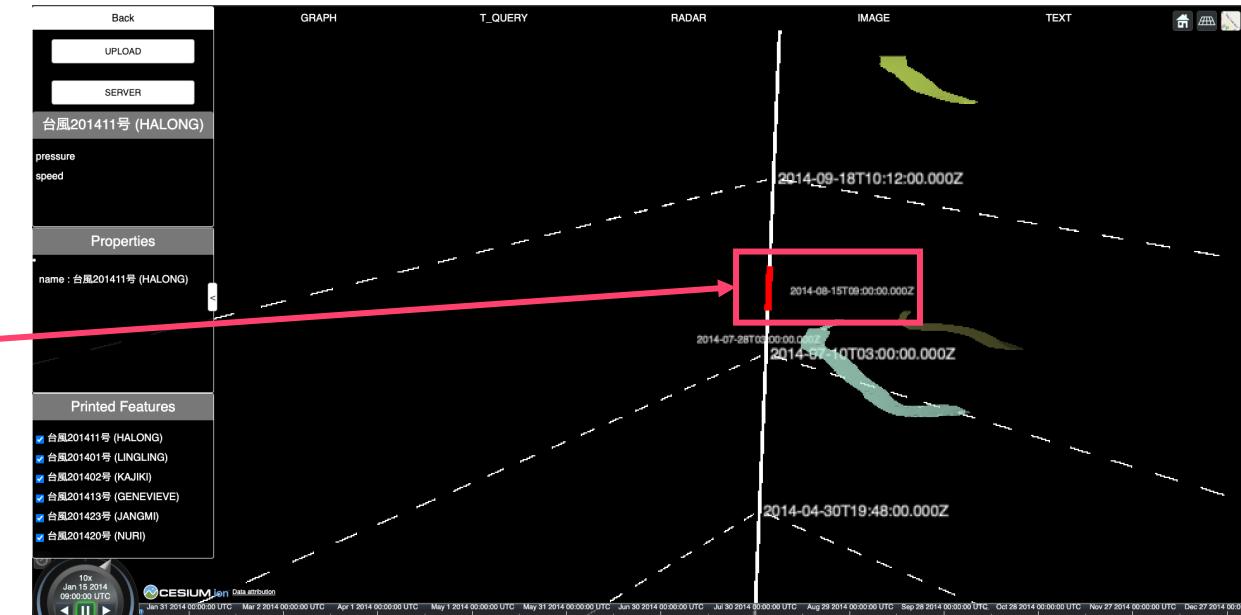
- Change the view modes (Static, Global, and Space-time cube view)

- Space-time cube view

- A map that converts the z-axis to a timeline instead of the z-coordinates of moving features
 - X and Y axis (location) + Z axis(time)
 - Easy to analyze objects moving at the same location
 - When clicking on a specific object, the time information for that object is displayed on the z-axis



Space-time View (Time based Map)

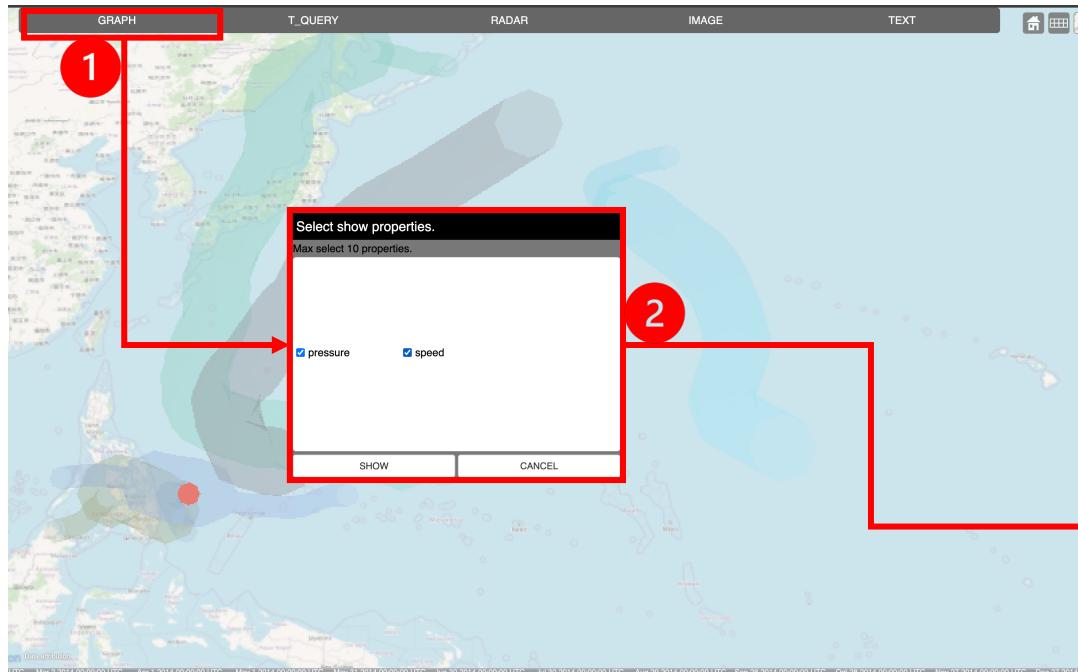


Display the time of selected MovingFeature

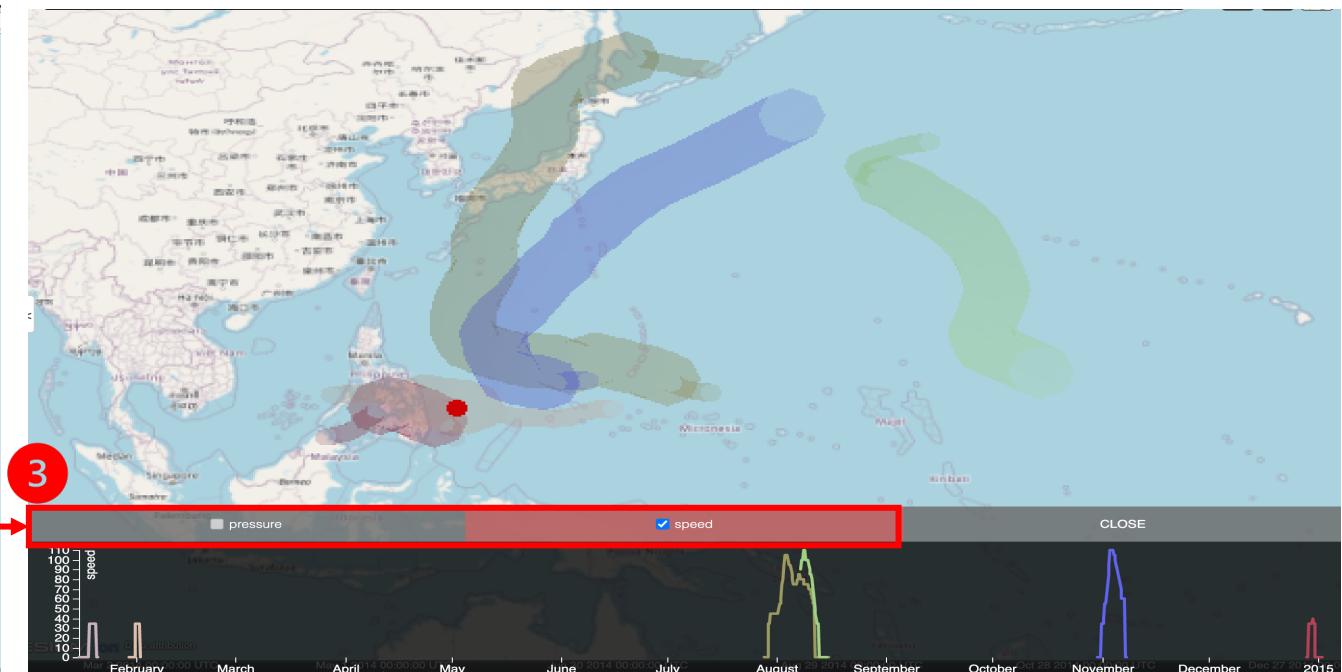
Practice with offline STINUUM

- **Visualize the temporal property (Measure type)**

- STINUUM supports graphing and analyzing the temporal properties of moving features.
 - 1) Display the list of properties included in the moving feature
 - 2) Select the one (or more) property
 - 3) Visualize property information as a graph



Select property to graph

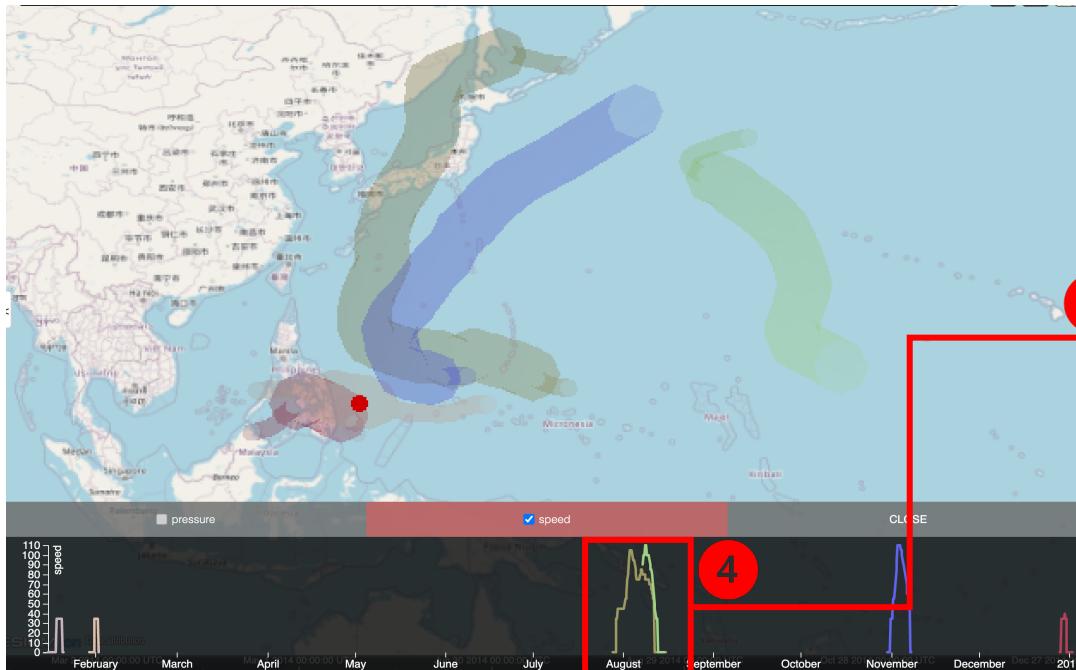


Display selected properties as graph

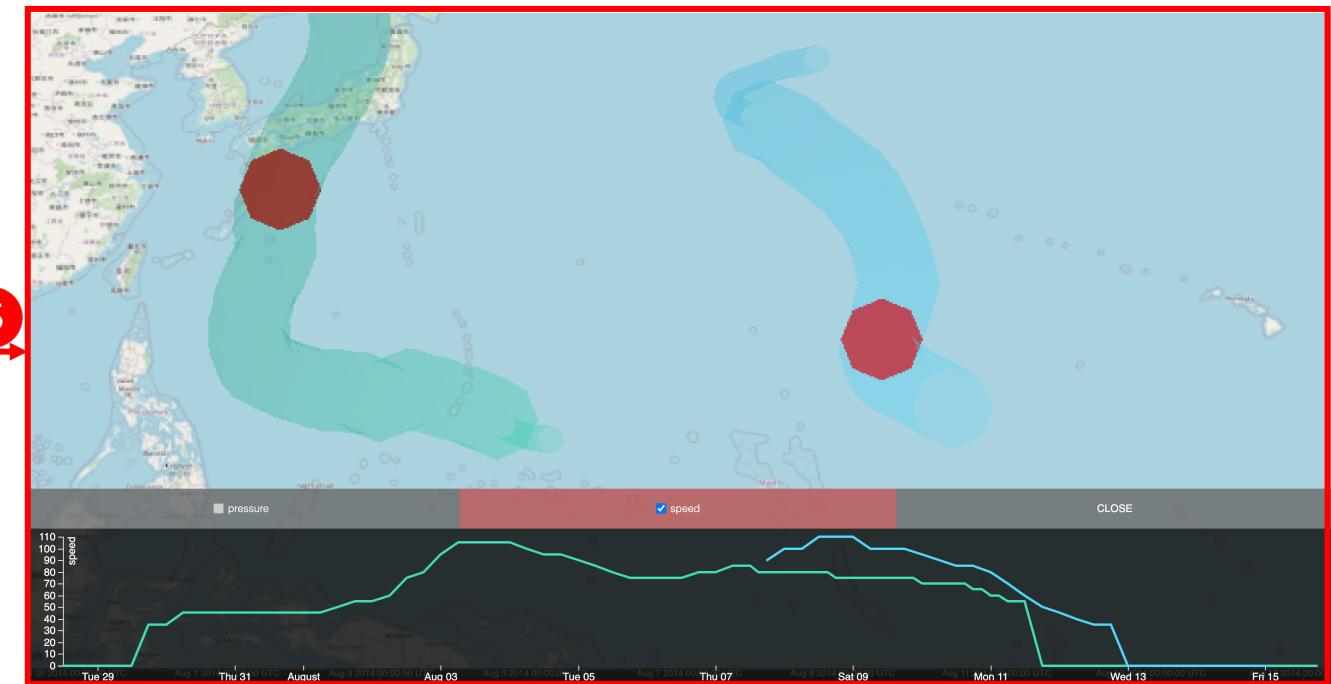
Practice with offline STINUUM

- **Visualize the temporal property (Measure type)**

- STINUUM supports graphing and analyzing the temporal properties of moving features.
- 4) Drag the specific area in Graph
 - 5) Display only the graph of the dragged area



Drag the specific area



Display the graph included the dragged area

Practice with offline STINUUM

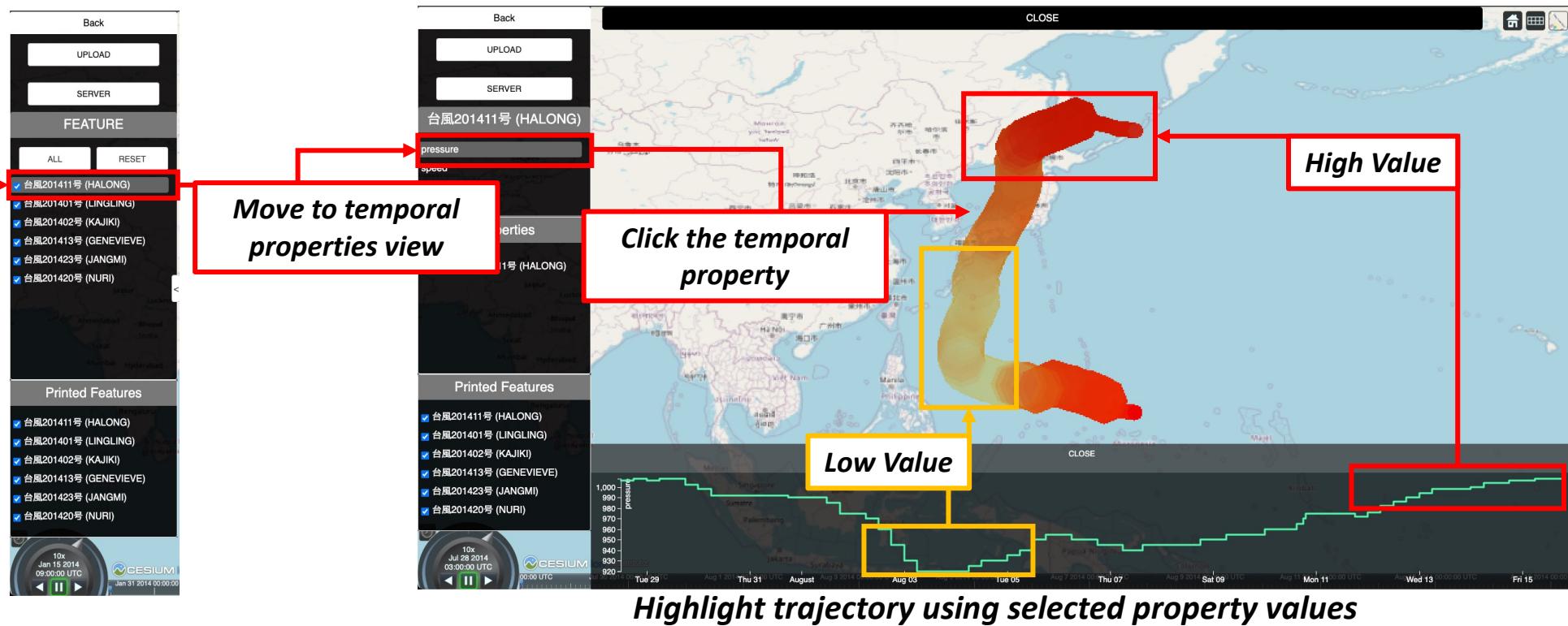
- **Visualize the Measure type with temporal geometry (highlight)**

- STINUUM supports the highlighting function with the temporal property
 - The color of the trajectory is represented by different colors depending on the value of the selected property

Low Value

High Value

- At the bottom, the selected property is displayed as a graph.

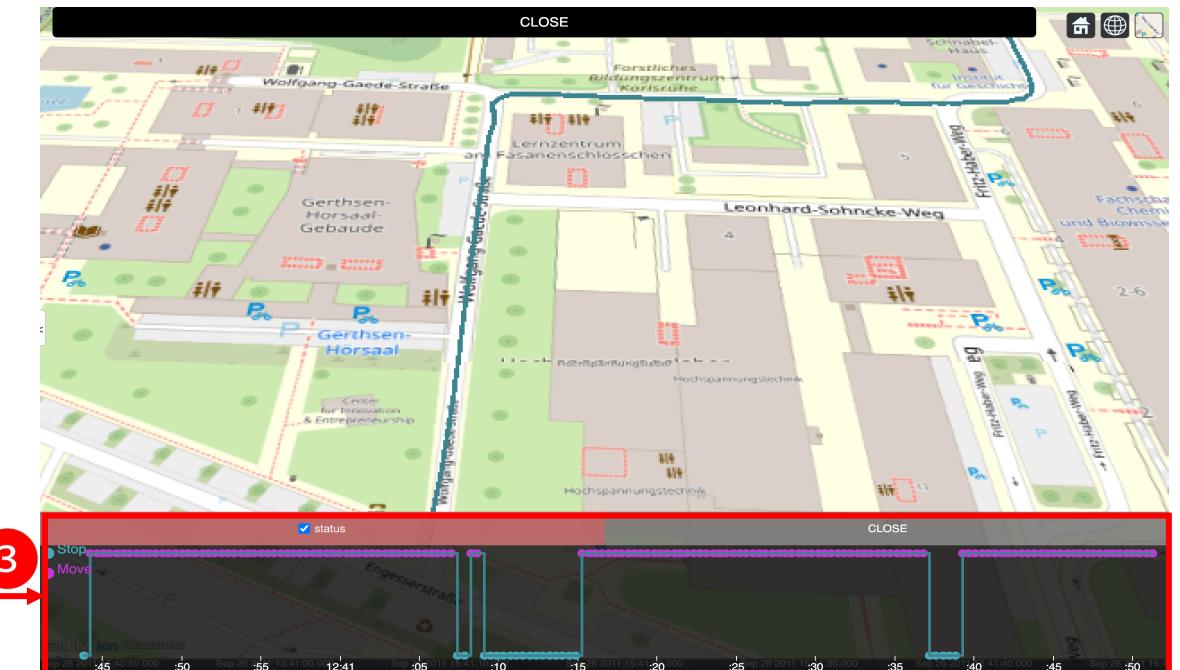
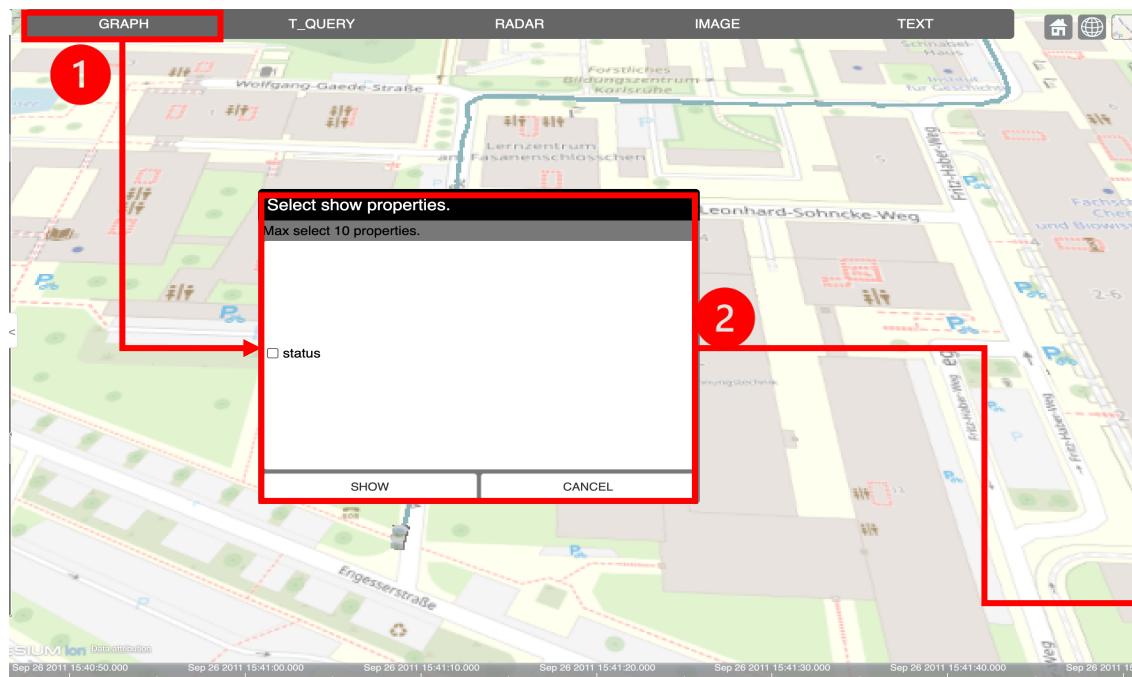


Practice with offline STINUUM

- Visualize the temporal property (Text type)

- Upload the new MF-JSON

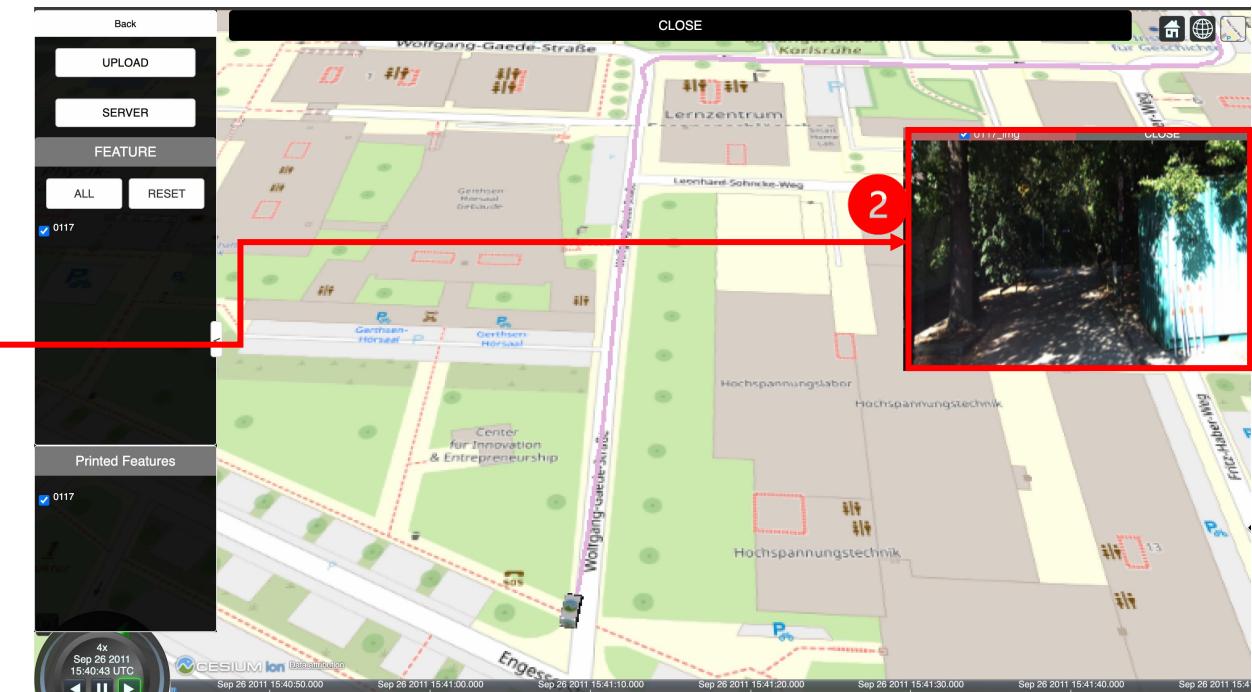
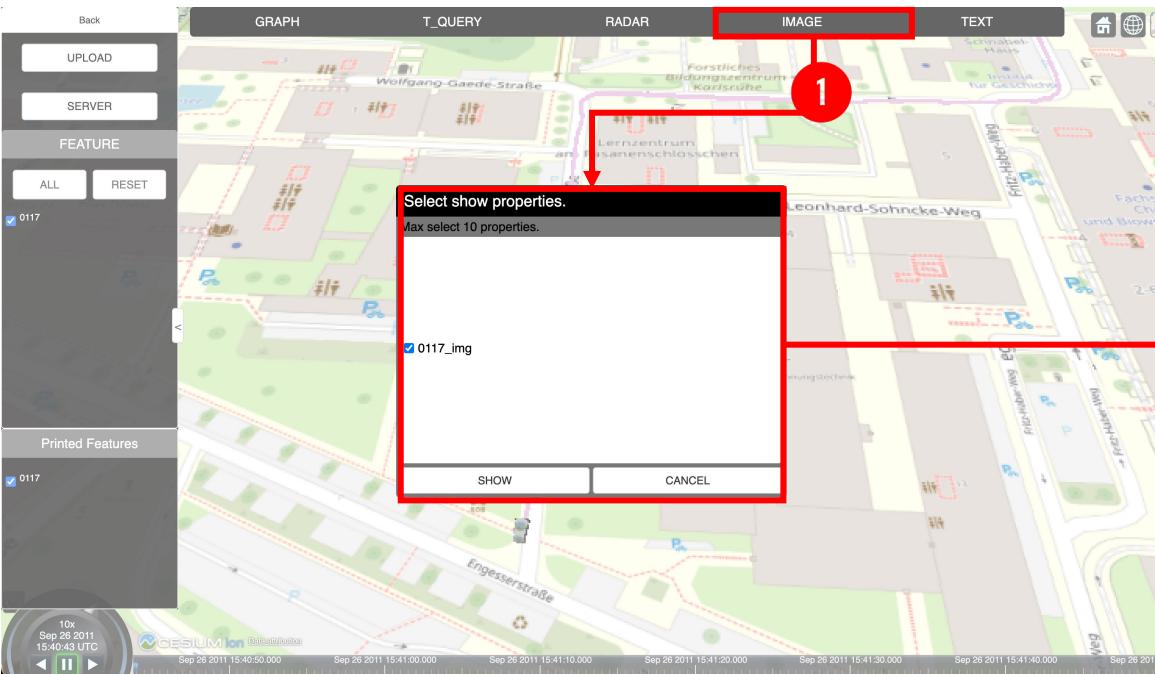
- Path of file: **Stinuum Web/data/MF-JSON Prism/PracticeDataSet/mfjson_point_car_with_image.json**
- 1) Display the list of properties included in the moving feature
 - 2) Select the one (or more) property
 - 3) Visualize property information as a graph



Practice with offline STINUUM

- **Visualize the temporal property (Image type)**

- STINUUM supports the visualization of image data with temporal properties.
 - All Images have time information
 - As time progresses in Cesium, the corresponding image is displayed



Practices with online STINUUM

1. Environment setup to connect with MF-API Server
2. Register MF-JSON data with MF-API-Handler
3. Get all MovingFeatureCollection in MF-API Server (/collections)
4. Get selected MovingFeatures (/collections/{c_id}/items)
5. Get and visualize the selected MovingFeature (/collections/{c_id}/items/{mf_id})
6. Get and visualize the TemporalGeometry (/collections/{c_id}/items/{mf_id}/tgsequence)
7. Get and visualize the TemporalProperties (/collections/{c_id}/items/{mf_id}/tProperties)

Practice with online STINUUM

- **Environment setup to connect with MF-API Server**

- Setting the server address and parameters related to the API-MF
 - Using **system.json** file in the project's root path
 - > Path of file: **Stinuum Web/system.json**
 - **mf_api_server_url**: MF-API Server address to connect (default localhost:8085)
 - **mf_limit**: number of moving features to import at once (default 10)
 - **mf_tg_limit**: number of temporal geometry to import at once (default 100)
 - **mf_tp_limit**: number of temporal properties to import at once (default 100)

Practice with online STINUUM

- **Register MF-JSON data with MF-API-Handler**

- How to use MF-API Handler

- It is created with Jupyter Notebook and can be executed in order to register data

- Connect to MF-API Server (Run Cell 2)

- Set the address of the MF-API server

- » `mf_api_server = "localhost:8085"`

- Register new MovingFeatureCollection (Run Cell 3)

- Setting the parameters required to register MovingFeatureCollection

- » `default_mf_collection_json = {
 "title": "Your_Collection_Title",
 "updateFrequency": 1000,
 "description": "Test"
}`

- Uploading the MF-JSON (Run Cell 4)

- Path of file: **Stinuum Web/data/MF-JSON Prism/Practice/mfjson_point_for_mf_api_server.json**

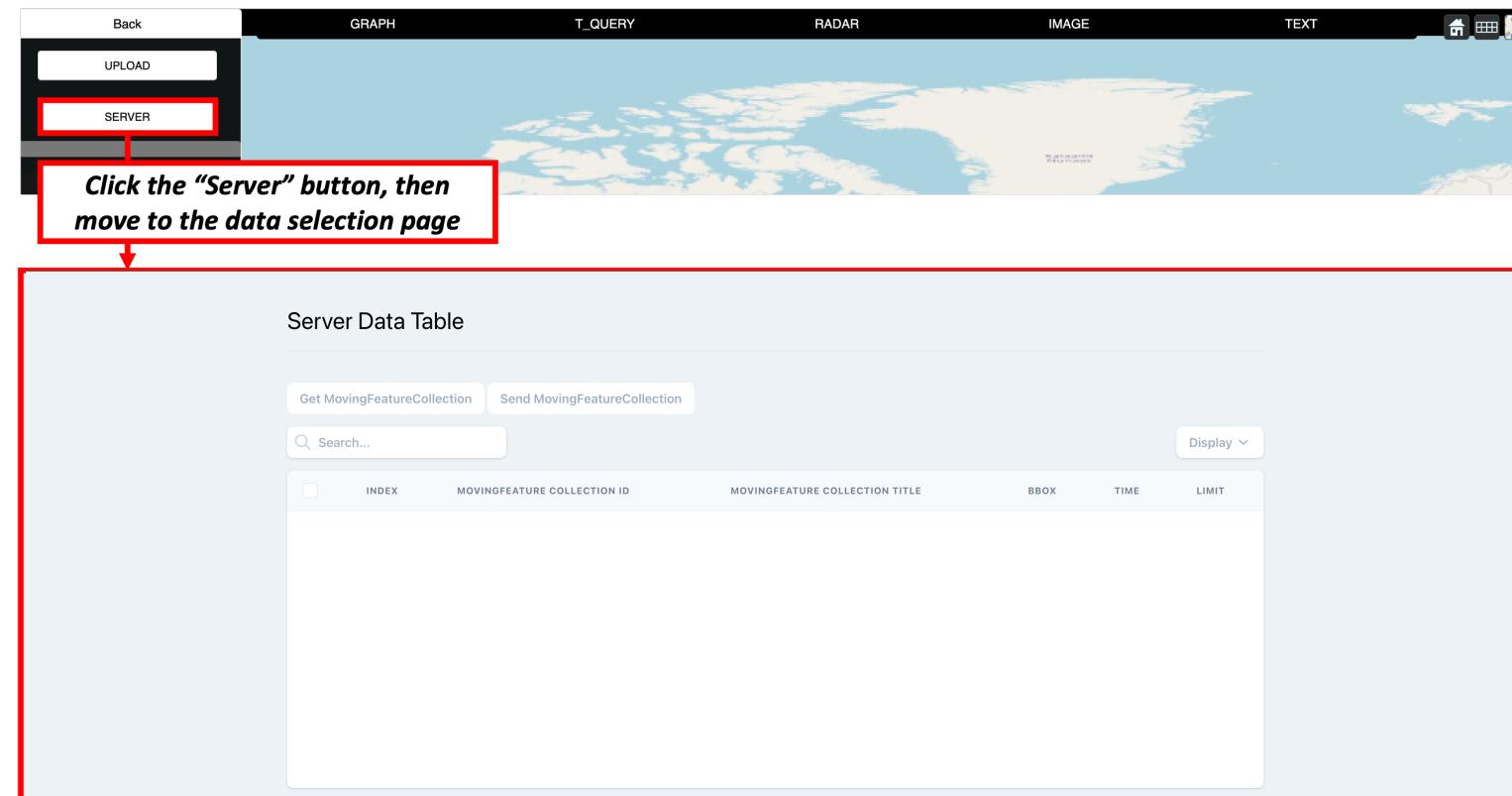
- Set the path to the file of the MF-JSON to be registered

- » `mf_json_path = "path_to_mf-json"`

Practice with online STINUUM

- Connect to MF-API Server from STINUUM

- STINUUM provides a connection function to the MF-API Server
 - 1) After running STINUUM, click the “**SERVER**” button in the menu on the left
 - 2) If a connection to the MF-API server is possible, move to the data selection page automatically
 - localhost:8080/dataSelect



Practice with online STINUUM

- Get all MovingFeatureCollection in MF-API server

- Users can get the **MovingFeatureCollection** from the MF-API Server
 - 1) Click the "*GET MovingFeatureCollection*" button
 - 2) Receive a list of **MovingFeatureCollection** from the MF-API Server
 - 3) If it succeeds, the result is added to the bottom table

Query Information	
GET	/collections

Server Data Table

Get MovingFeatureCollection → **Get the MovingFeatureCollection**

Search... Display ▾

INDEX	MOVINGFEATURE COLLECTION ID	MOVINGFEATURE COLLECTION TITLE	BBOX
0	f46b4f01-800e-4e05-b5d9-f0455db2b3bc	GeoJsonTest2	8.3397745066324,49.010726498382,1,8.42328878
1	d5a8046b-b14b-47fb-ad3f-3112063843ac	GeoJsonTest3	8.4118378284011,49.011668355047,1,8.413889836
2	3fd46e33-8b9e-4546-973d-d3199edb233a	moving_feature_collection_sample	8.4228904357119,49.010726498382,1,8.42328878
3	f59cc431-756d-45b5-a1af-08d160a7dc2c	GeoJsonTest8	8.4228904357119,49.010726498382,1,8.42328878
4	358dfcba-8104-403a-a370-4835a1e0d140	GeoJsonTest7	8.4228904357119,49.010726498382,1,8.42328878
5	49392923-4ab9-4af9-a2c9-cd56a922c25a	GeoJsonTest4	4.265321832093116,50.78614171809345,4.4767712
6	f7369a4a-4fa6-44ef-ae05-890b5a2763f6	GeoJsonTest4	

Result of getting MovingFeatureCollection

Practice with online STINUUM

- Send the MovingFeatureCollection data to the main page
 - Select one or more rows that users need to get the MovingFeatureCollection
 - And then click the "*Send MovingFeatureCollection*" button to sends the selected column values to the main page
 - localhost:8080/main

1 rows are selected

Send the selected MovingFeatureCollection data

Get MovingFeatureCollection **Send MovingFeatureCollection**

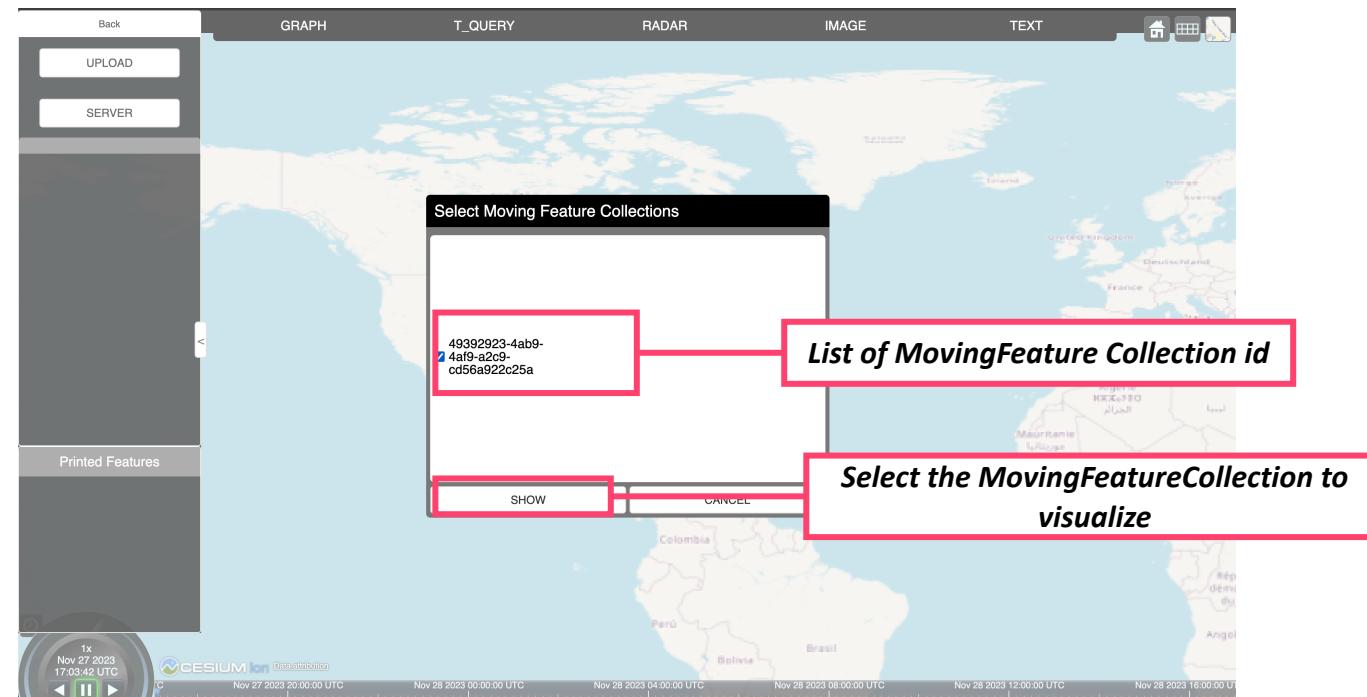
Search... Display ▾

<input type="checkbox"/>	INDEX	MOVINGFEATURE COLLECTION ID	MOVINGFEATURE COLLECTION TITLE	BBOX
<input type="checkbox"/>	0	f46b4f01-800e-4e05-b5d9-f0455db2b3bc	GeoJsonTest2	8.3397745066324,49.010726498382,1,8.42328878
<input type="checkbox"/>	1	d5a8046b-b14b-47fb-ad3f-3112063843ac	GeoJsonTest3	8.4118378284011,49.011668355047,1,8.413889836
<input type="checkbox"/>	2	3fd46e33-8b9e-4546-		8.3904357119,49.010726498382,1,8.42328878
<input type="checkbox"/>	3	f59cc431-756d-45b5-		8.3904357119,49.010726498382,1,8.42328878
<input type="checkbox"/>	4	358dfcba-8104-403a-a370-4835a1e0d140	GeoJsonTest7	8.4228904357119,49.010726498382,1,8.42328878
<input checked="" type="checkbox"/>	5	49392923-4ab9-4af9-a2c9-cd56a922c25a	GeoJsonTest4	4.265321832093116,50.78614171809345,4.476771,
<input type="checkbox"/>	6	f7369a4a-4fa6-44ef-ae05-890b5a2763f6	GeoJsonTest4	

Practice with online STINUUM

- Get the **MovingFeature** list from the selected **MovingFeatureCollection**
 - STINUUM displays a list of **MovingFeatureCollection** IDs from the MF-API Server
 - 1) Select one or more IDs and then click the "**SHOW**" button to get the **MovingFeatureCollection** data
 - 2) Add the **MovingFeatureCollection** data to the left menu (**Collection Layer**)
 - 3) Click on an ID displayed in the collection layer, the included **MovingFeature** information is visualized

→ From now on, users can use STINUUM's features to analyze moving features

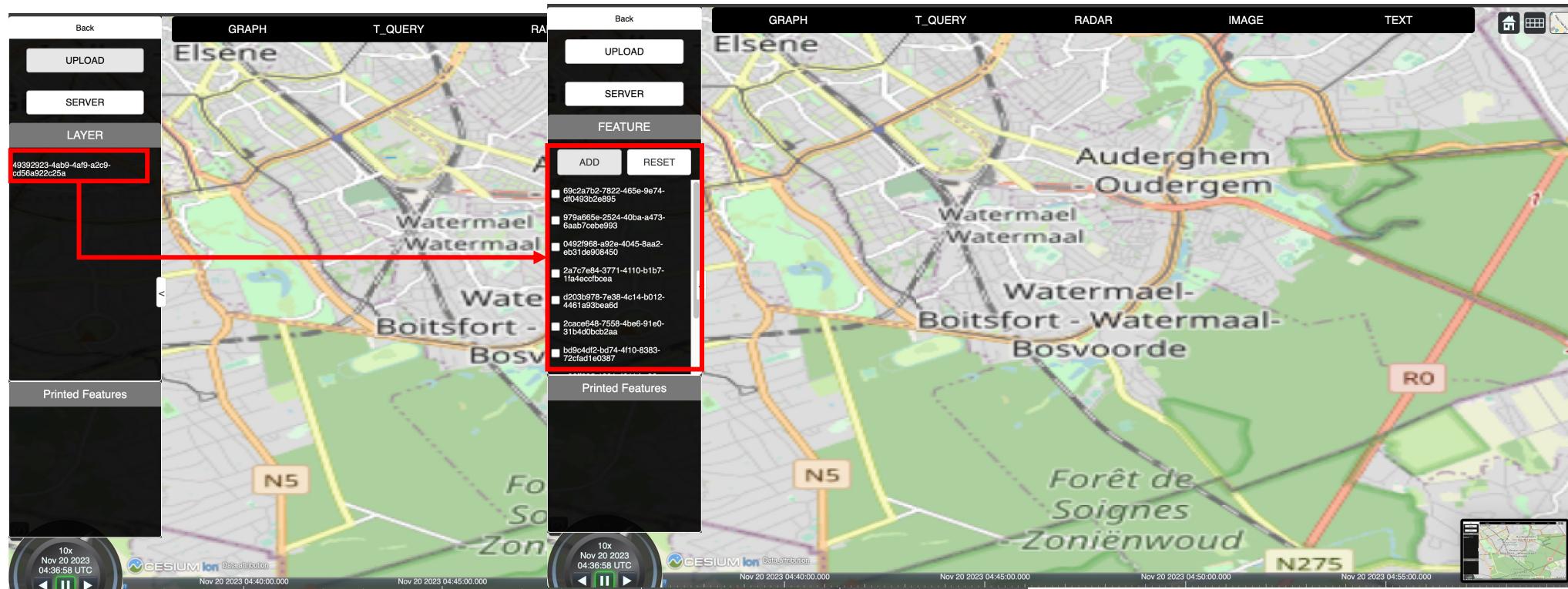


Practice with online STINUUM

• Get selected MovingFeatures

- Click a **MovingFeatureCollection** item in the Collection Layer
- Get each **MovingFeature** information from the MF-API Server

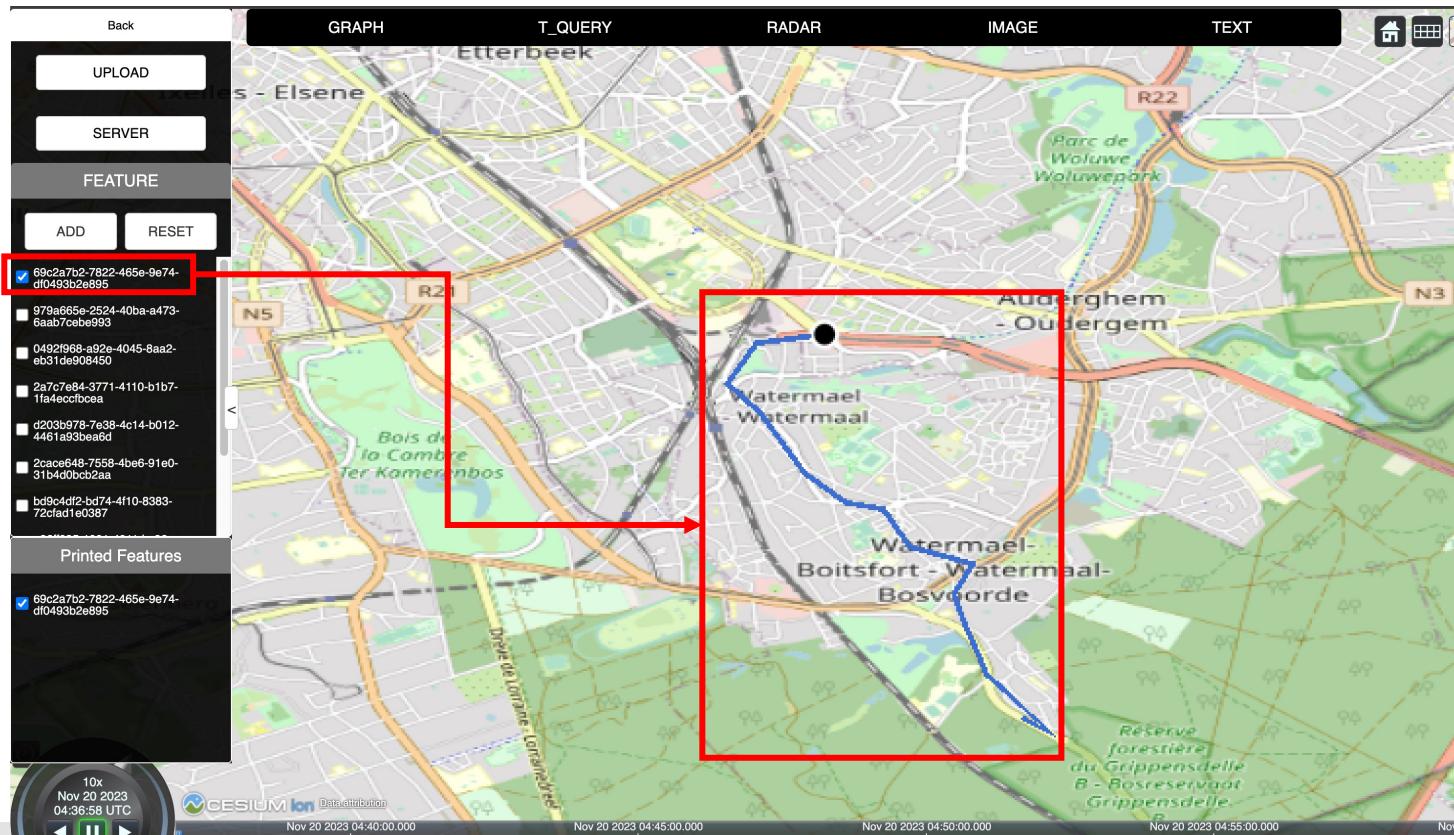
Query Information	
GET	/collections/{c_id}/items
c_id	ID of Selected Moving Feature Collection
Parameters information	
limit	mf_limit (default 10)
datetime	Datetime of selected Moving Feature Collection



Practice with online STINUUM

- Get and visualize the TemporalGeometry

- Click the check box of **MovingFeature** from the Feature Layer
- Get the **TemporalGeometry** data from the MF-API Server using MF-API

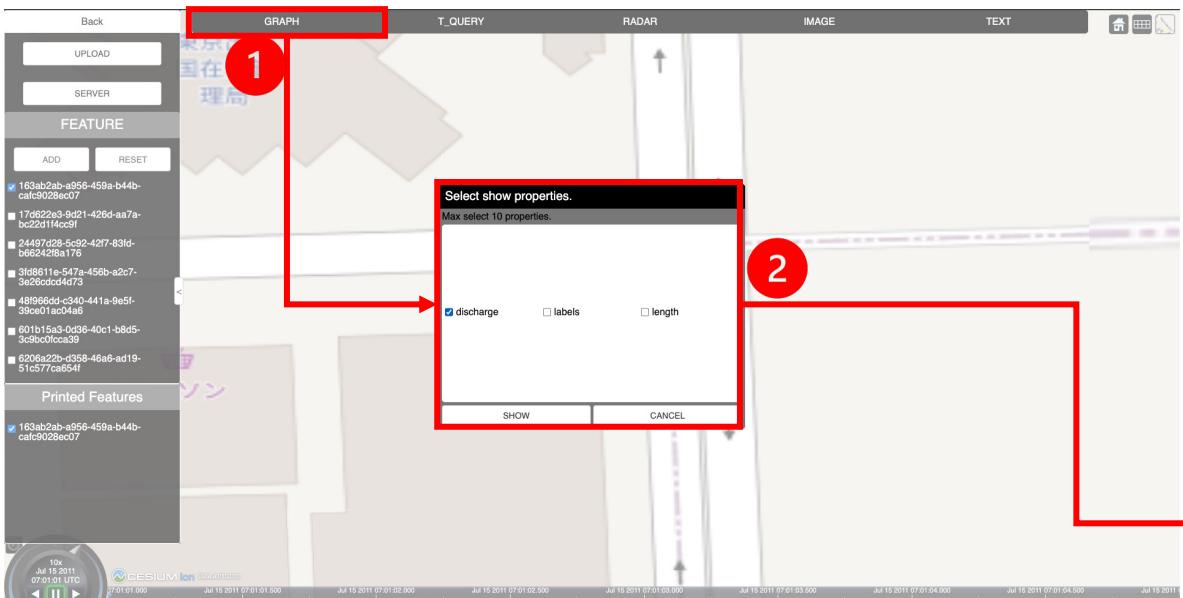


Query Information	
GET	/collections/{c_id}/items/mf_id/tgsequence
c_id	ID of Selected Moving Feature Collection
mf_id	ID of Selected Moving Feature
Parameters information	
limit	mf_tg_limit (default 100)
datetime	Datetime of selected Moving Feature

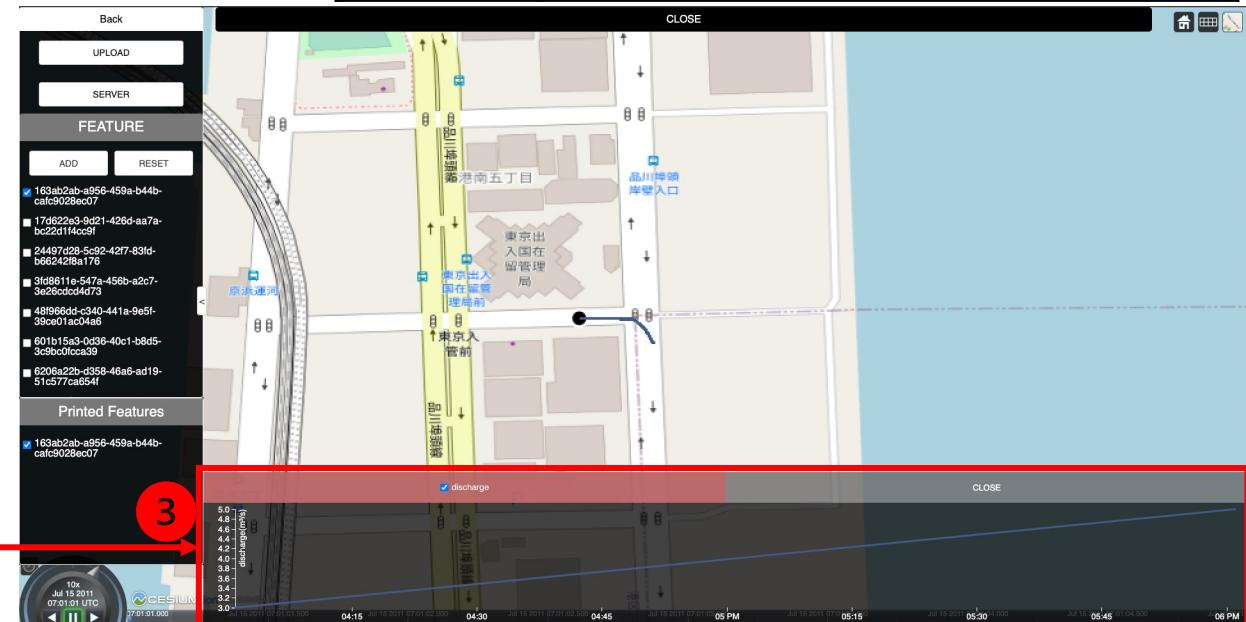
Practice with online STINUUM

• Get and visualize the TemporalProperties

- 1) Click the **GRAPH** button: Display the list of temporal property included in the **MovingFeature**
- 2) Select a temporal property
- 3) Visualize temporal property data as a graph



Query Information	
GET	/collections/{c_id}/items/mf_id/tproperties
c_id	ID of Selected Moving Feature Collection
mf_id	ID of Selected Moving Feature
Parameters information	
limit	mf_tp_limit (default 100)
datetime	Datetime of selected Moving Feature



Thank you