# How to implement OGC API – Moving Features with pygeoapi and MobilityDB

***Wijae Cho***, Taehoon Kim, Kyoungsook Kim

***Technical Staff***

***Artificial Intelligence Research Center***

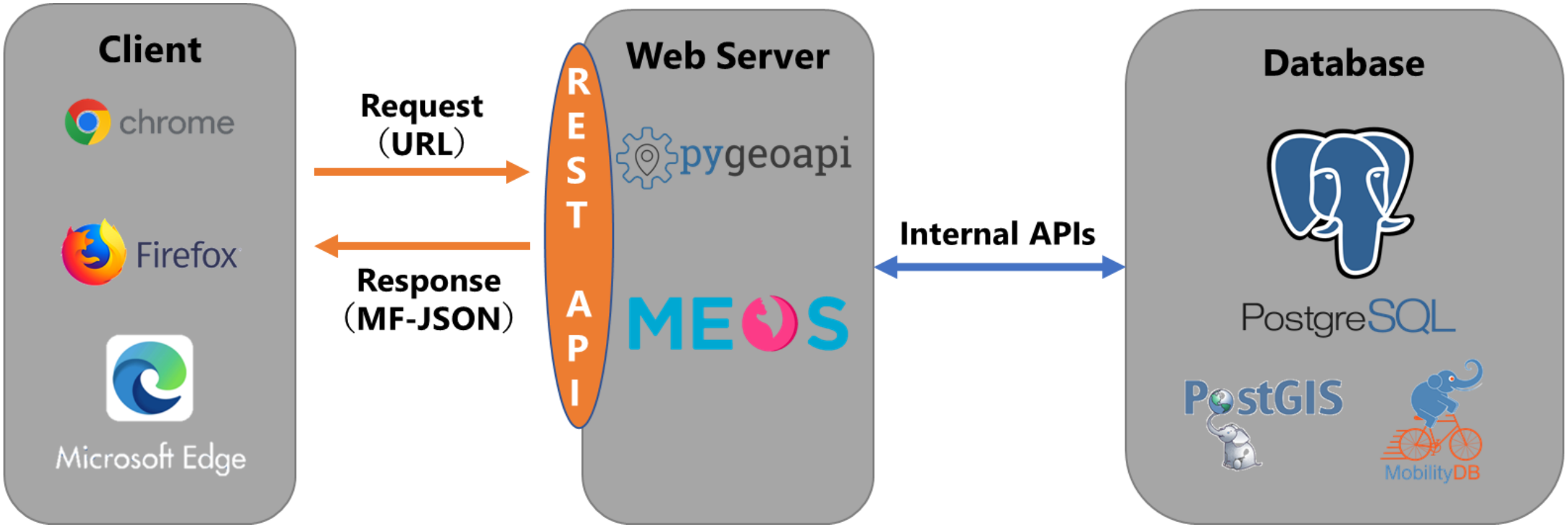
***National Institute of Advanced Industrial Science and Technology***

***(AIST)***

# A brief explanation of pygeoapi and install it

1. Overall architecture
2. What is pygeoapi
3. How to install (using Docker)

# Overall architecture



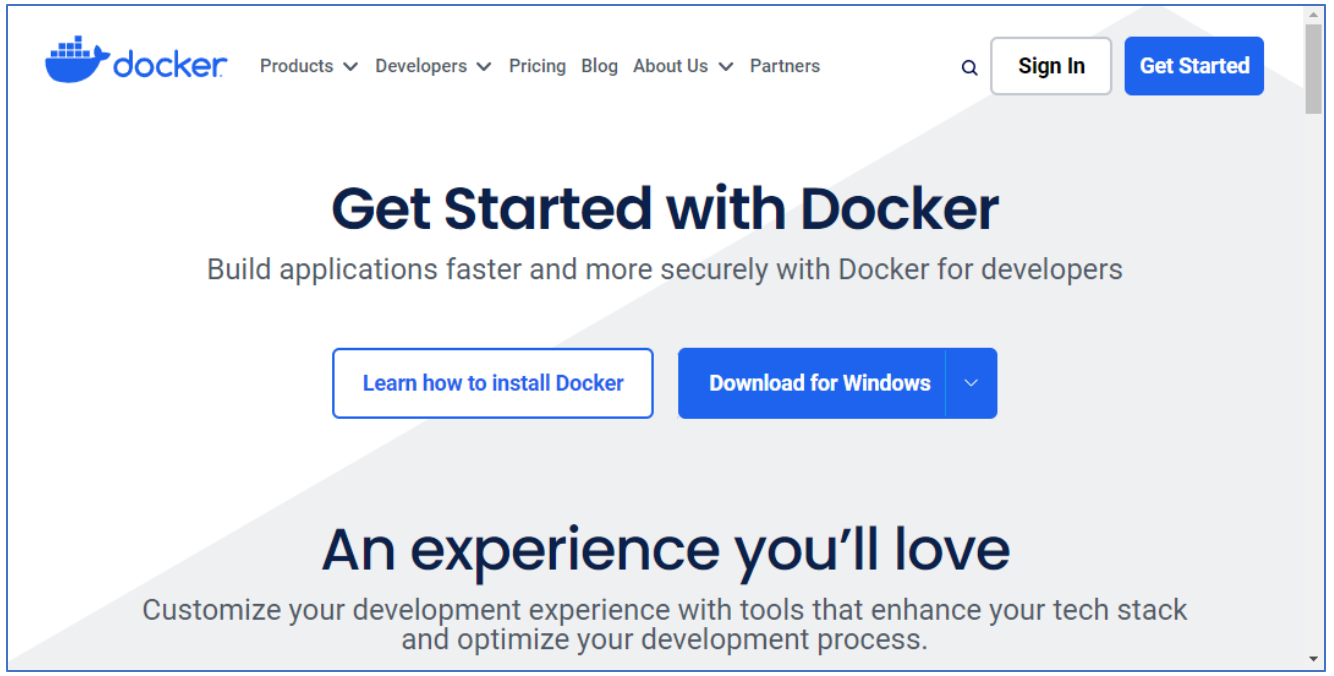
# What is pygeoapi

- Python server implementation of the OGC API standard
  - Providing the ability to deploy RESTful OGC API endpoints using OpenAPI, GeoJSON, and HTML
- Features
  - Easy-to-install, ready-to-use, state-of-the-art OGC API server
  - Easy-to-use OpenAPI/Swagger for developers
  - Built on a robust plugin framework Custom data connections, formats, and processes can be built

[pygeoapi] https://pygeoapi.io/

# How to install (using Docker)

- Install docker
  - https://www.docker.com/get_started/



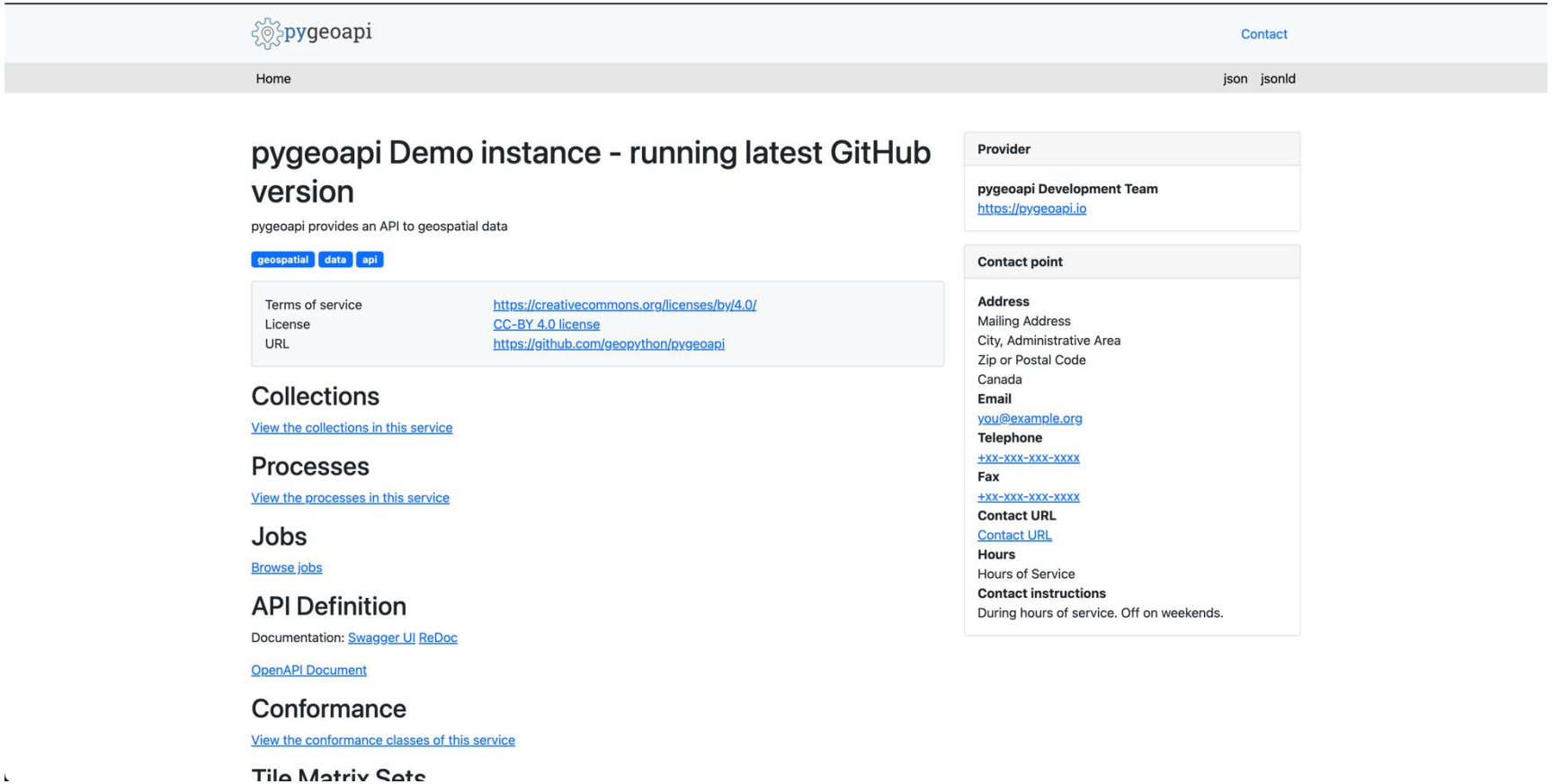
# How to install (using Docker)

- Pull Docker image
  - Executing command below at command-line

- Optional
  - Checking the status of docker application

# How to install (using Docker)

- Checking the localhost
  - http://localhost:5050/



# A brief explanation of MobilityDB

1. What is MobilityDB
2. Process flow from MovingFeature to MobilityDB
3. Process flow from TemporalGeometry to MobilityDB
4. Process flow from TemporalProperties to MobilityDB

# What is MobilityDB

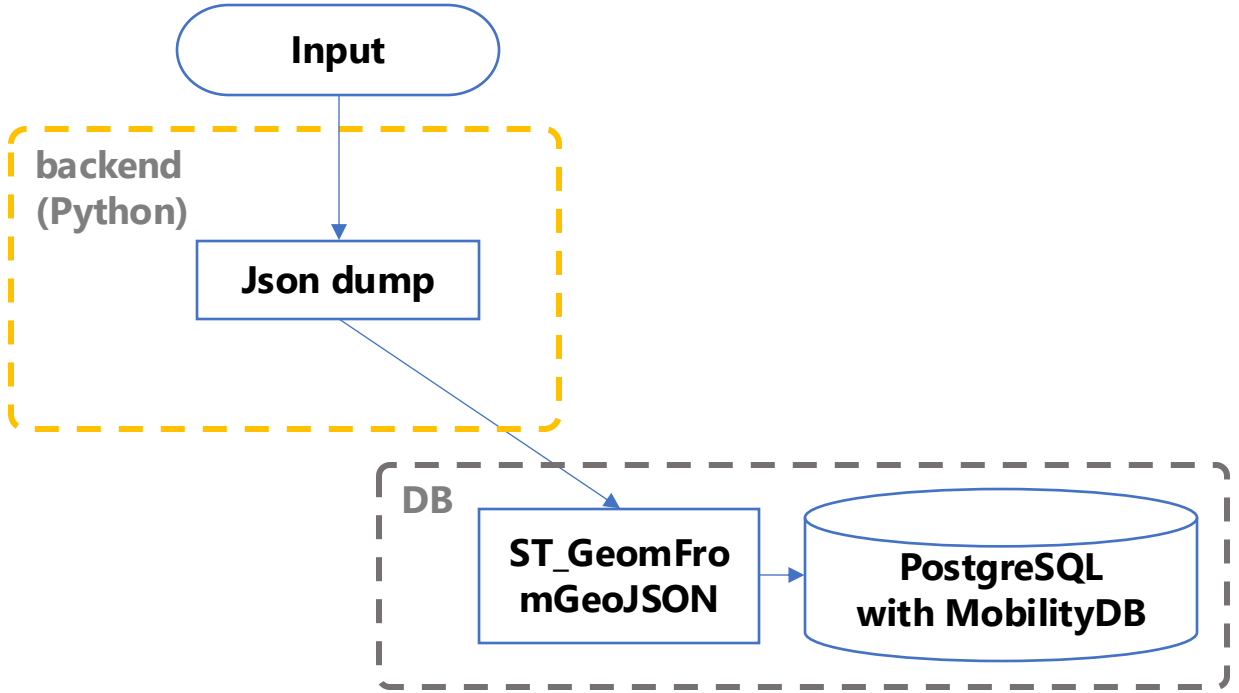
- Postgres extensions for geospatial traces of moving objects such as GPS traces
- To the PostgreSQL database and its spatial extension PostGIS, Spatiotemporal object support has been added
- Example of MobilityDB specific types:
  - **tgeogpoint, tfloat, ttext, etc…**


[MobilityDB] https://github.com/MobilityDB/MobilityDB

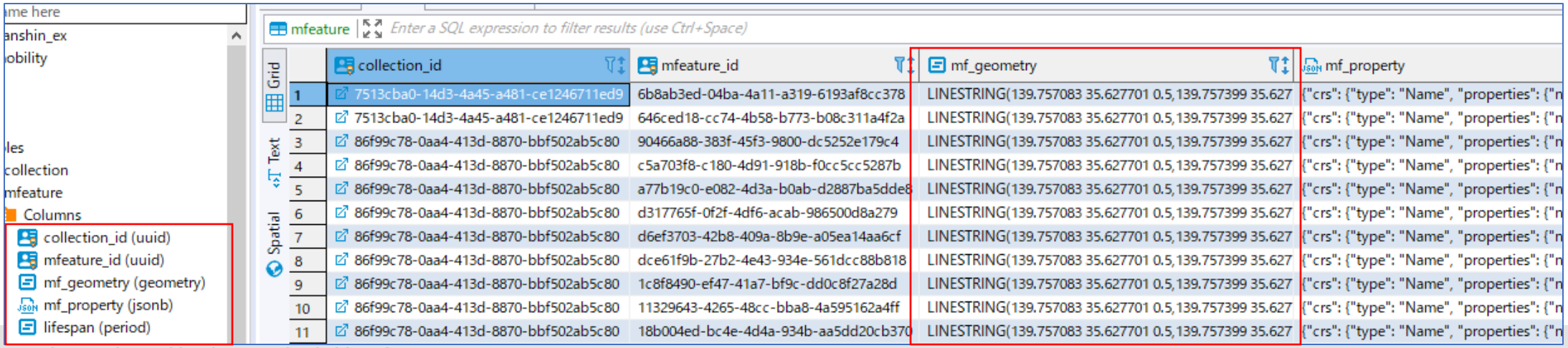
# What is MobilityDB

- Functions used to handle
  - Postgres (PostGIS)
    - MovingFeature Geometry -> ST_GeomFromGeoJSON
    - GeoJSON response -> st_asgeojson()
  - Postgres (MobilityDB)
    - subTrajectory parameter -> tstzspan()
    - leaf parameter -> tstzset()
    - bbox parameter -> stboxx(), stboxz()
    - temporalGeometry, temporalProperty GET request -> extent()
  - PyMEOS
    - MFJSON interconversion -> from_mfjson(), as_mfjson()

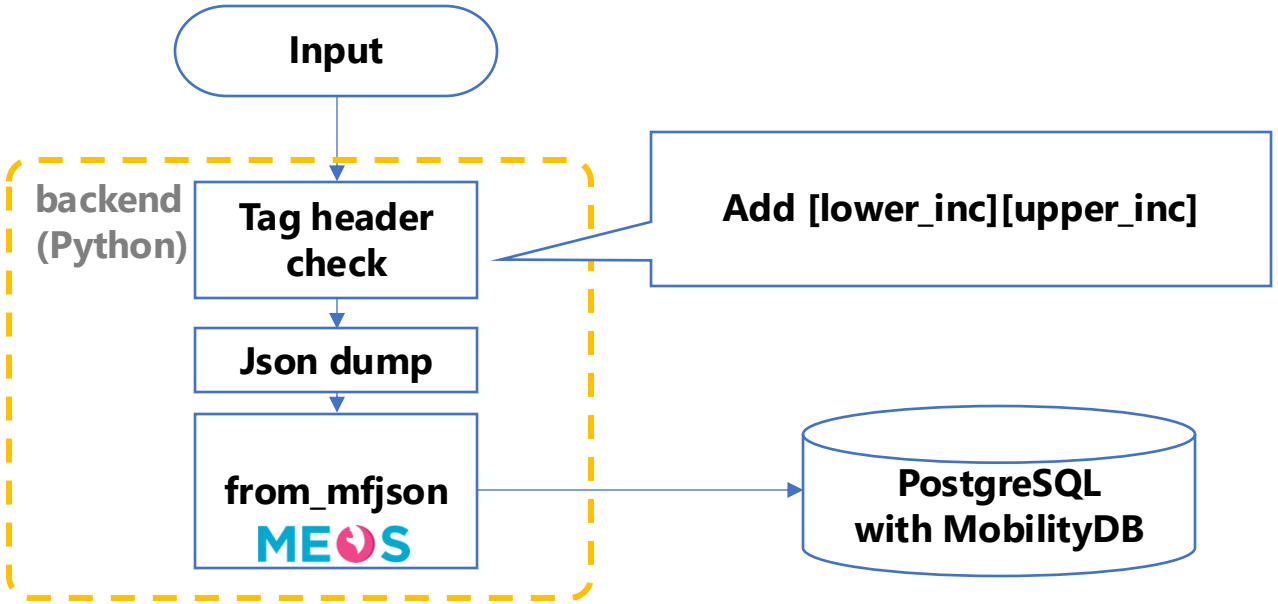
# Process flow from MovingFeature to MobilityDB



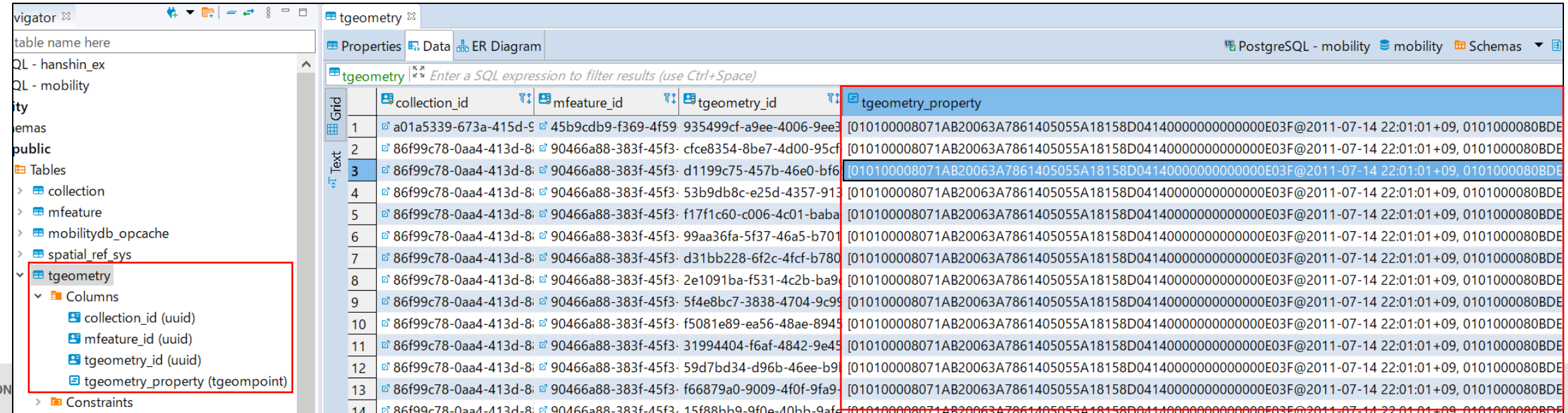

**[mfeature]**



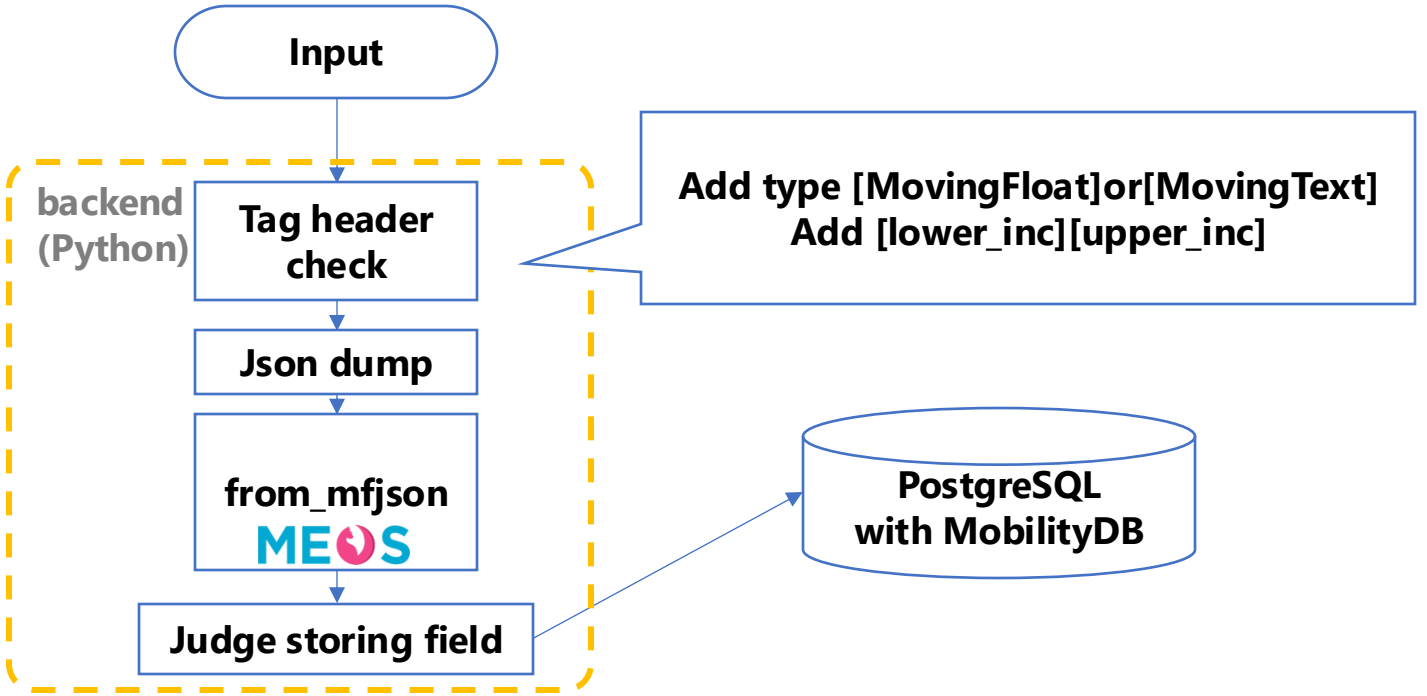
# Process flow from TemporalGeometry to MobilityDB



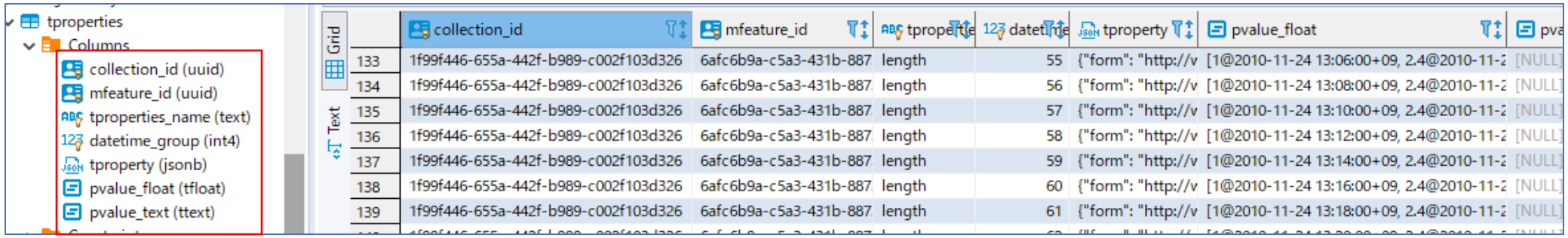

**[tgeometry]**



# Process flow from TemporalProperties to MobilityDB




## [tproperties]

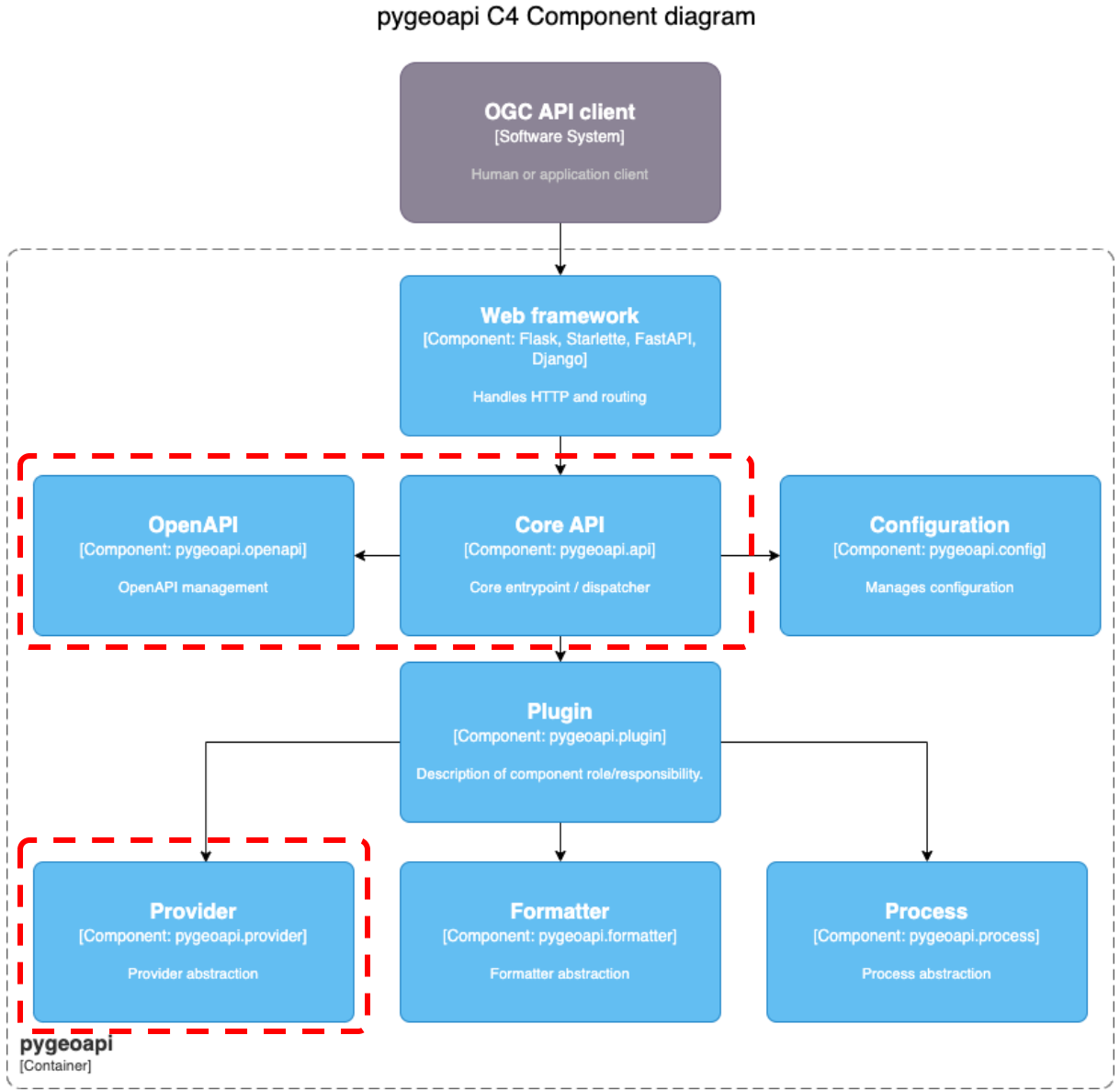


# How to extend pygeoapi to support OGC API – Moving Features

1. Extension parts of pygeoapi
2. Process flow in pygeoapi
3. Used libraries for extension structure
4. PyMEOS in the process flow

# Extension parts of pygeoapi

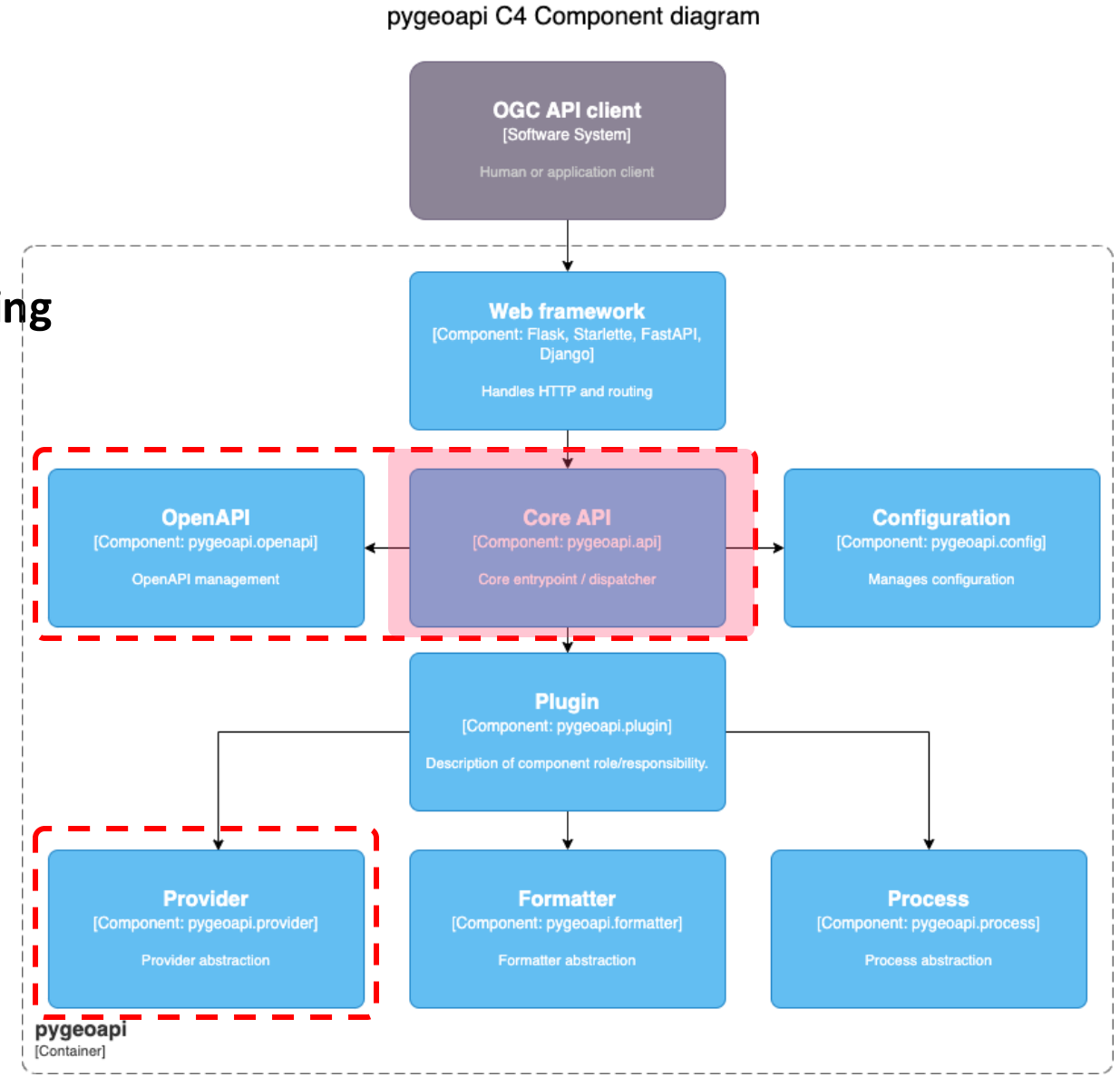
Extension




[pygeoapi] https://docs.pygeoapi.io/en/stable/how-pygeoapi-works.html

# Extension parts of pygeoapi

- In Core API
  - pygeoapi/api/__init__py
  - pygeoapi/api/movingfeature.py
  - **Parameter identification and response formatting**

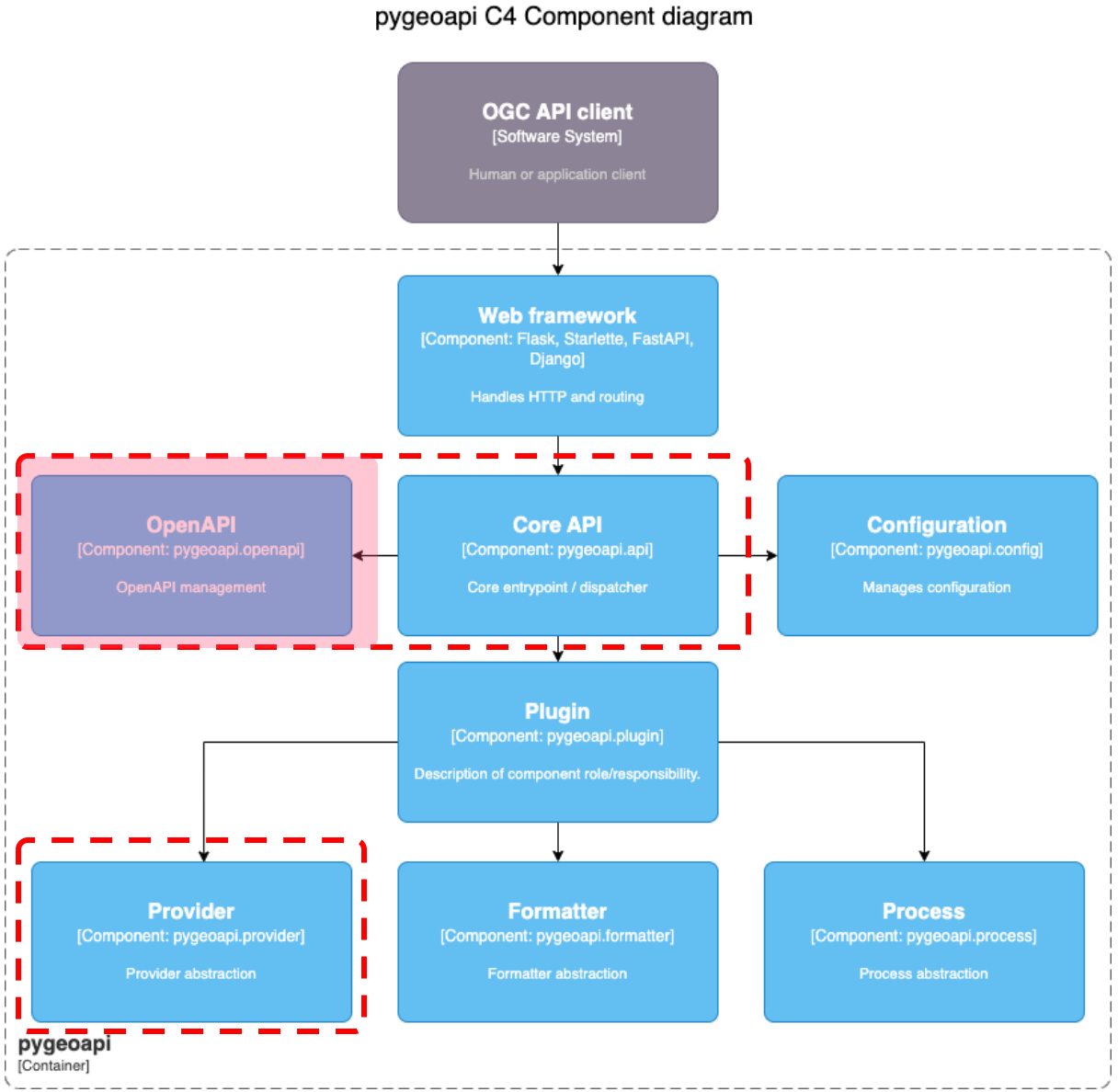



Extension

[pygeoapi] https://docs.pygeoapi.io/en/stable/how-pygeoapi-works.html

# Extension parts of pygeoapi

- In OpenAPI
  - pygeoapi/flask_app.py
  - pygeoapi/openapi.py
  - **Determine processing from Swagger requests**

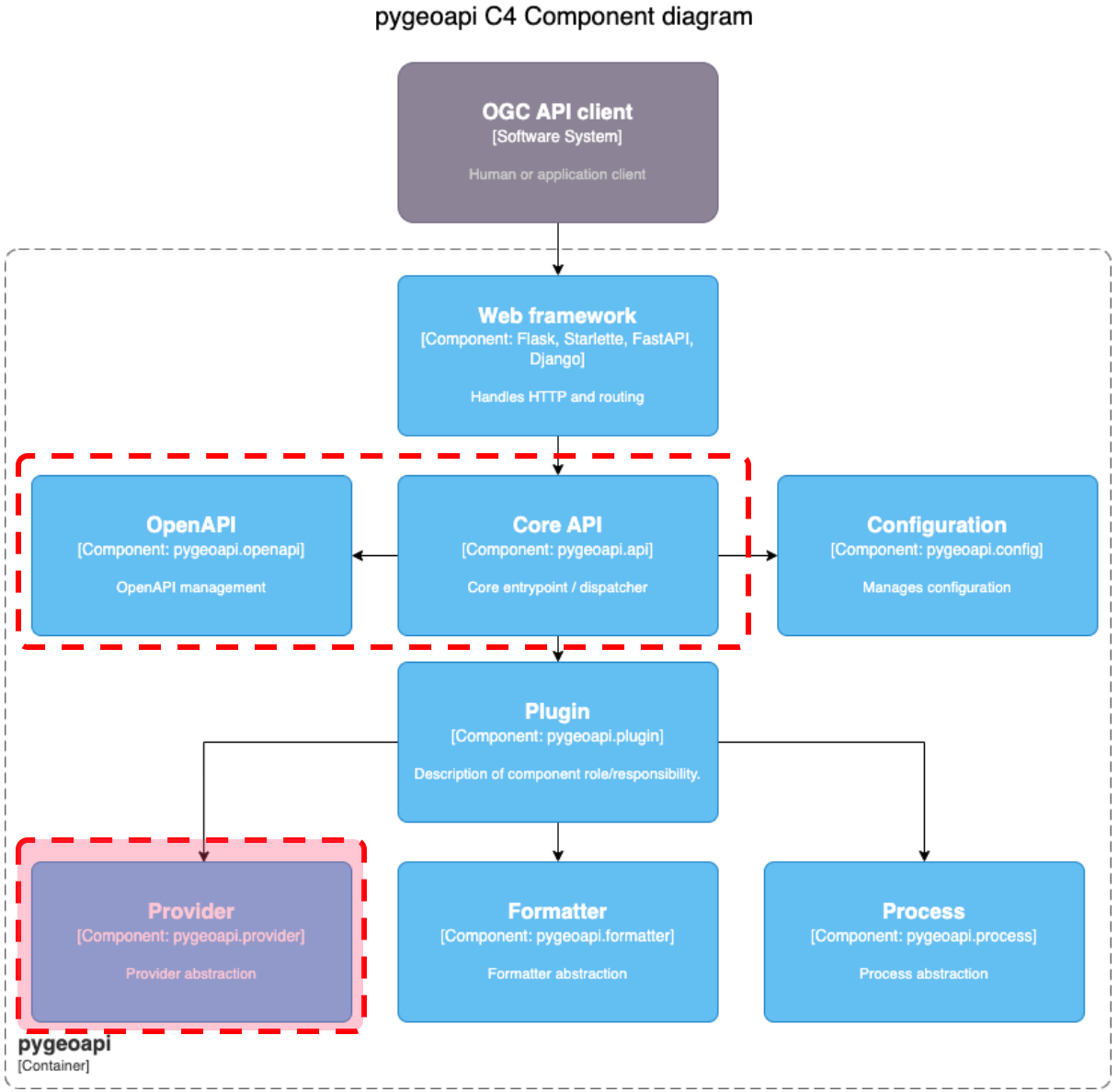



Extension

[pygeoapi] https://docs.pygeoapi.io/en/stable/how-pygeoapi-works.html

# Extension parts of pygeoapi

- In Provider
  - pygeoapi/provider/postgresql_mobilitydb.py
  - **Query Execution**
  - **SQL query generation from parameters**



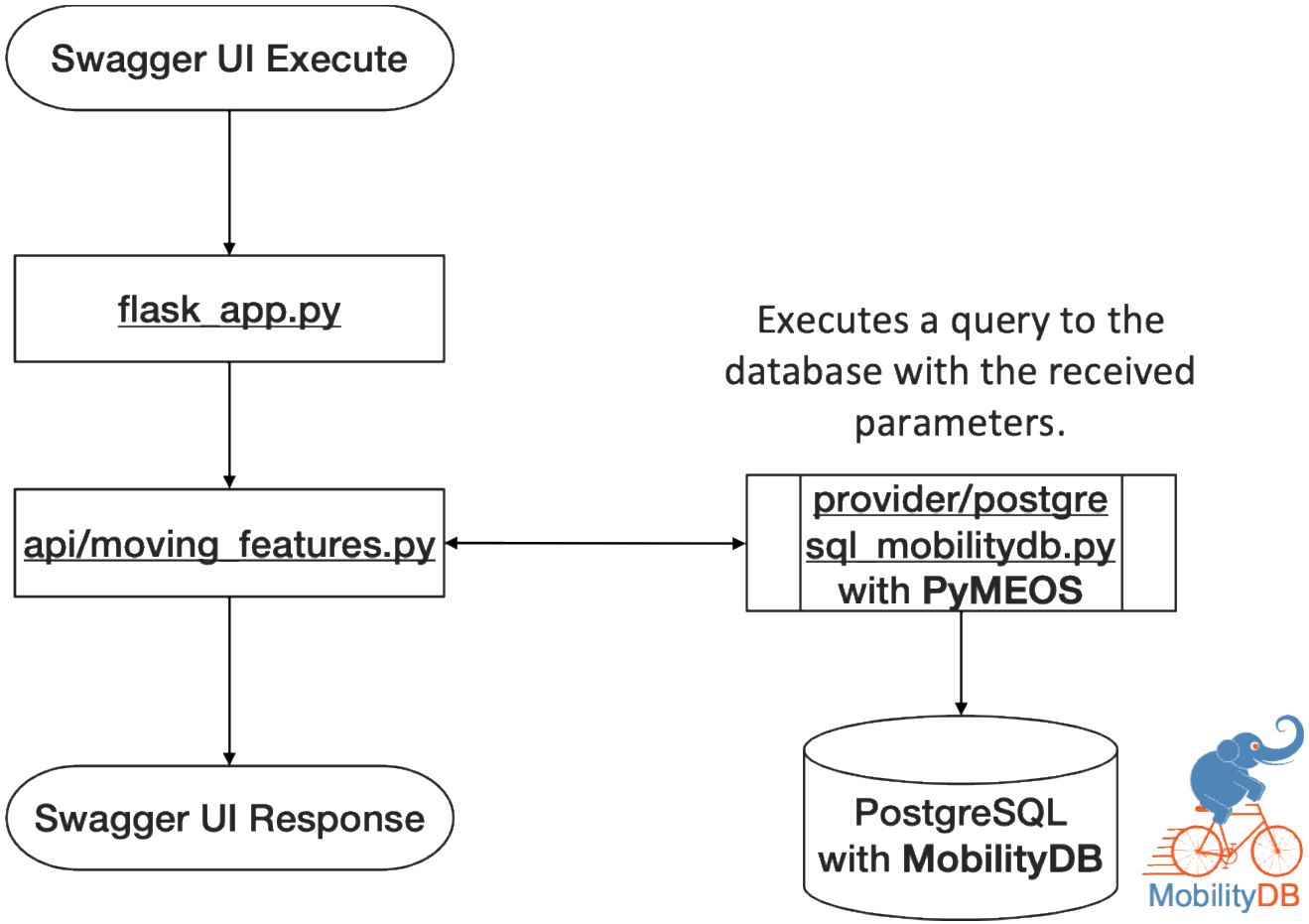

Extension

[pygeoapi] https://docs.pygeoapi.io/en/stable/how-pygeoapi-works.html

# Process flow in pygeoapi

Process requests such as URI and GET, PUT, POST, DELETE operation with parameters

It handles parameters and receives the result of the query from the database.




Executes a query to the database with the received parameters.

[MF-API] https://github.com/aistairc/mf-api

# Used libraries for extension structure

- GeoAlchemy2
  - To convert query paramerters
    - **ST_MakeEnvelope, ST_3DMakeBox, ST_MakePoint**
- psycopg2
  - Postgres database adapter for python
- SQLAlchemy
  - Python SQL toolkit
- Postgres extension
  - uuid-ossp: generating universally unique identifies (UUID)

[GeoAlchemy] https://geoalchemy-2.readthedocs.io/en/latest/
[psycopg2] https://pypi.org/project/psycopg2
[SQLAlchemy] https://www.sqlalchemy.org/
[uuid-ossp] https://www.postgresql.org/docs/current/uuid-ossp.html

# Used libraries for extension structure

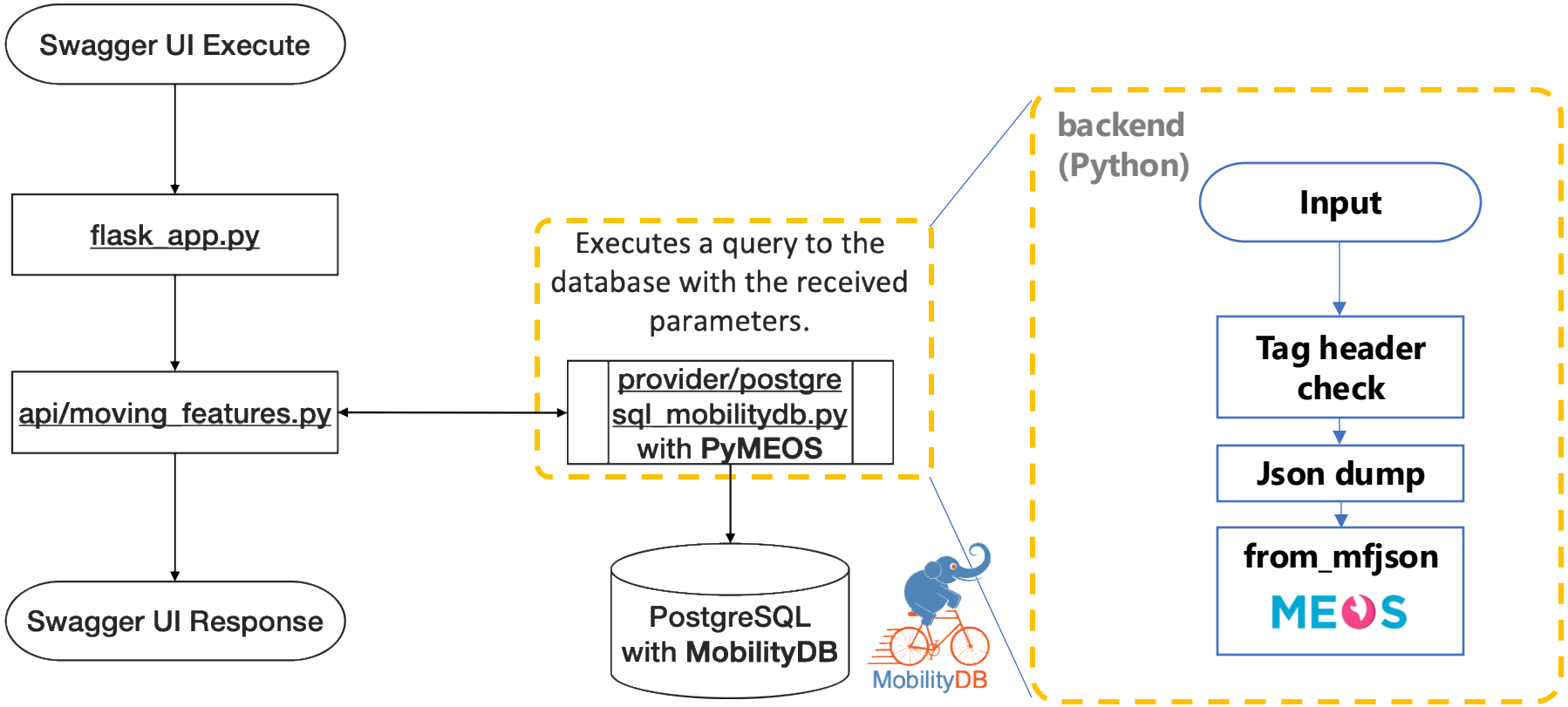
- PyMEOS (Mobility Engine, Open Source)
  - PyMEOS is a library built to top of MEOS that provides all its functionality wrapped in a set of Python class
  - MEOS
    - C library which enables the manipulation of temporal and spatio-temporal data based on MobilityDB's data types and functions

[pymeos] https://github.com/MobilityDB/PyMEOS

# PyMEOS in the process flow

Process requests such as URI and GET, PUT, POST, DELETE operation with parameters

It handles parameters and receives the result of the query from the database.



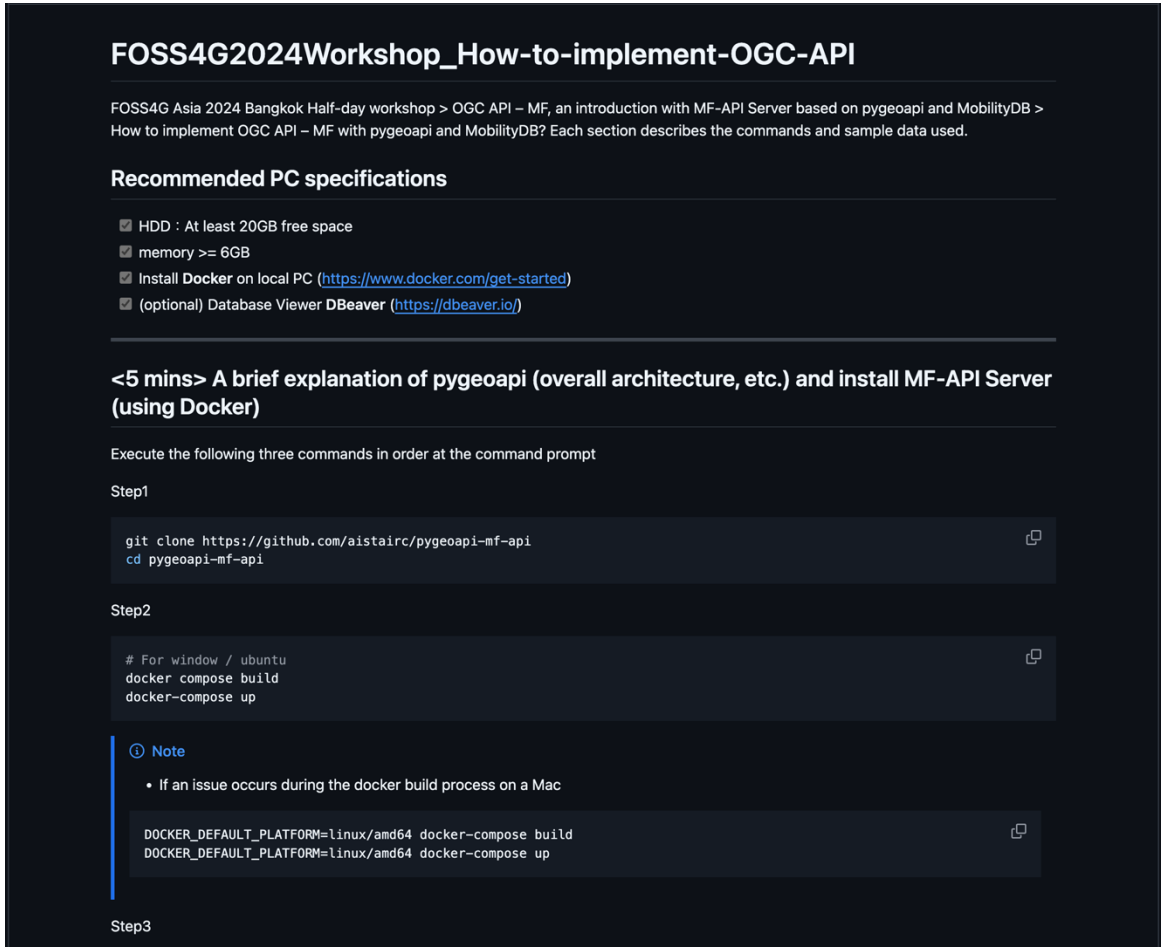

[pymeos] https://github.com/MobilityDB/PyMEOS

# Verify that each API works properly using Swagger

1. Workshop materials
2. APIs about MovingFeatureCollection
3. APIs about MovingFeatures
4. APIs about TemproalGeometry
5. APIs about TemporalProperties
6. APIs about TemporalProperty
7. APIs about TemporalGeometryQuery

# Workshop materials

- URL: https://github.com/opengeospatial/mf-swg/tree/foss4g_asia_2024
  - Sharing sample data with command for practicing



## FOSS4G2024Workshop_How-to-implement-OGC-API

FOSS4G Asia 2024 Bangkok Half-day workshop > OGC API – MF, an introduction with MF-API Server based on pygeoapi and MobilityDB > How to implement OGC API – MF with pygeoapi and MobilityDB? Each section describes the commands and sample data used.

### Recommended PC specifications

- ☑ HDD：At least 20GB free space
- ☑ memory >= 6GB
- ☑ Install **Docker** on local PC (https://www.docker.com/get-started)
- ☑ (optional) Database Viewer **DBeaver** (https://dbeaver.io/)

### <5 mins> A brief explanation of pygeoapi (overall architecture, etc.) and install MF-API Server (using Docker)

Execute the following three commands in order at the command prompt

Step1

```
git clone https://github.com/aistairc/pygeoapi-mf-api
cd pygeoapi-mf-api
```

Step2

```
# For window / ubuntu
docker compose build
docker-compose up
```

Note

- If an issue occurs during the docker build process on a Mac

```
DOCKER_DEFAULT_PLATFORM=linux/amd64 docker-compose build
DOCKER_DEFAULT_PLATFORM=linux/amd64 docker-compose up
```

Step3

# APIs about MovingFeatureCollection

## POST /collections

No1


**Parameters**

Request body (application/json)

Reference GitHub Page

```
{
  "title": "moving_feature_collection_sample1",
  "updateFrequency": 1000,
  "description": "FOSS4G2024 Asia Workshop”,
  "itemType": "movingfeature"
}
```

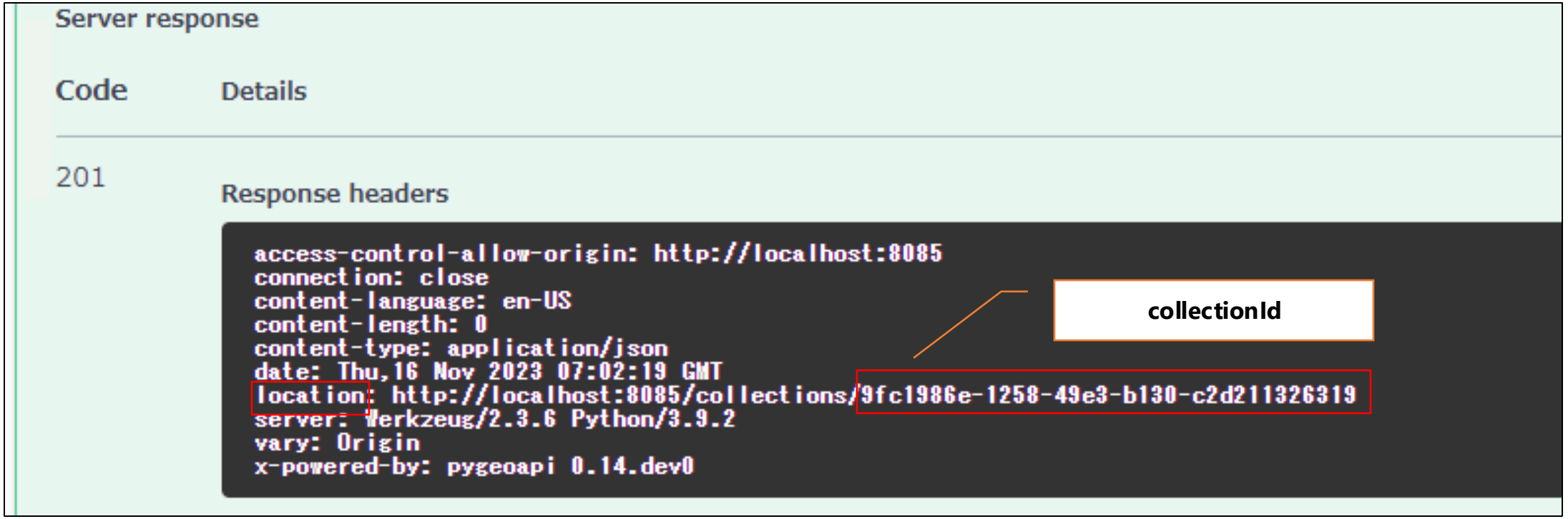
# APIs about MovingFeatureCollection

AIST
Create the Future, Collaborate Together

## POST /collections

No1

## Result

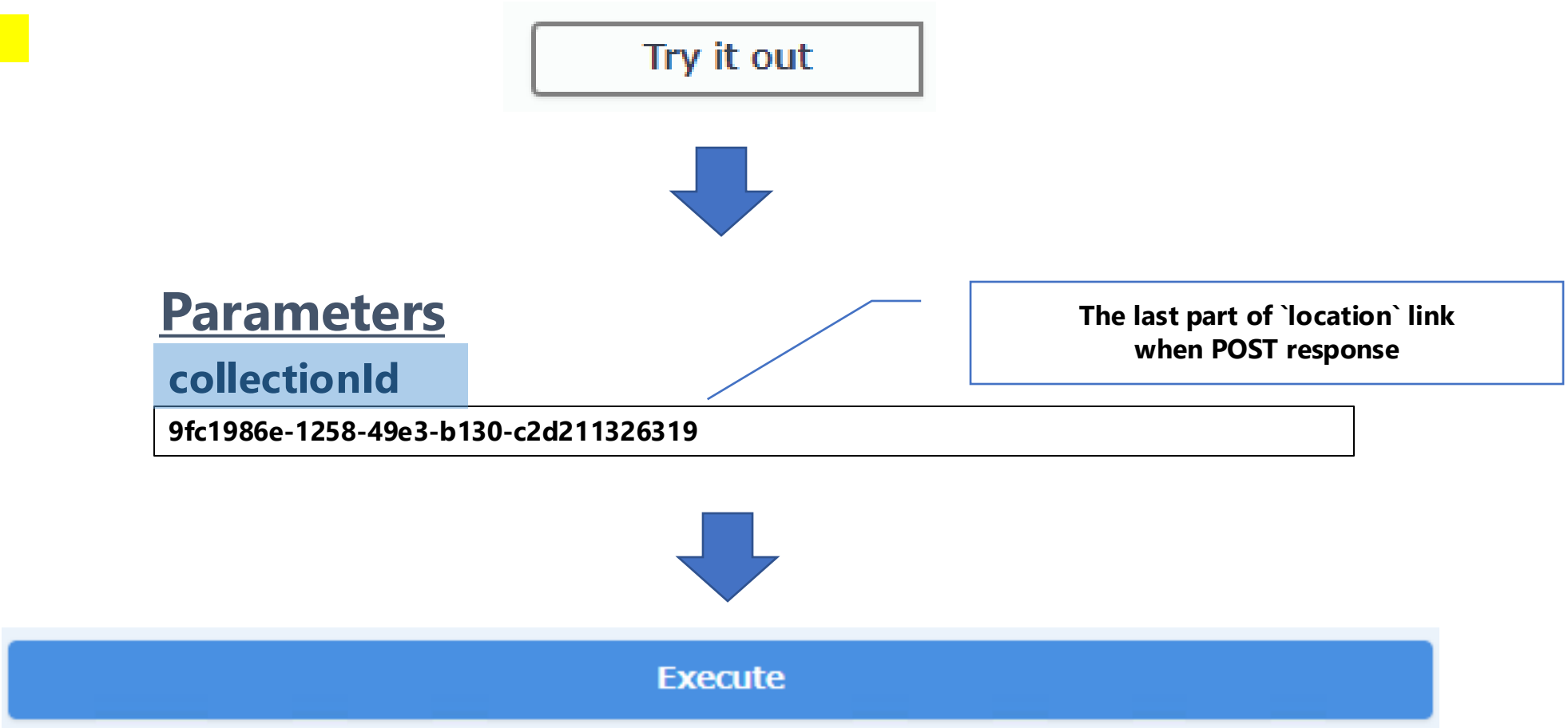



**collectionId : 9fc1986e-1258-49e3-b130-c2d211326319**

# APIs about MovingFeatureCollection

## GET /collections/{collectionId}

No2



# APIs about MovingFeatureCollection

## GET /collections/{collectionId}

**No2**

**Result**

Server response

| Code | Details |
| --- | --- |
| 200 | Response body |

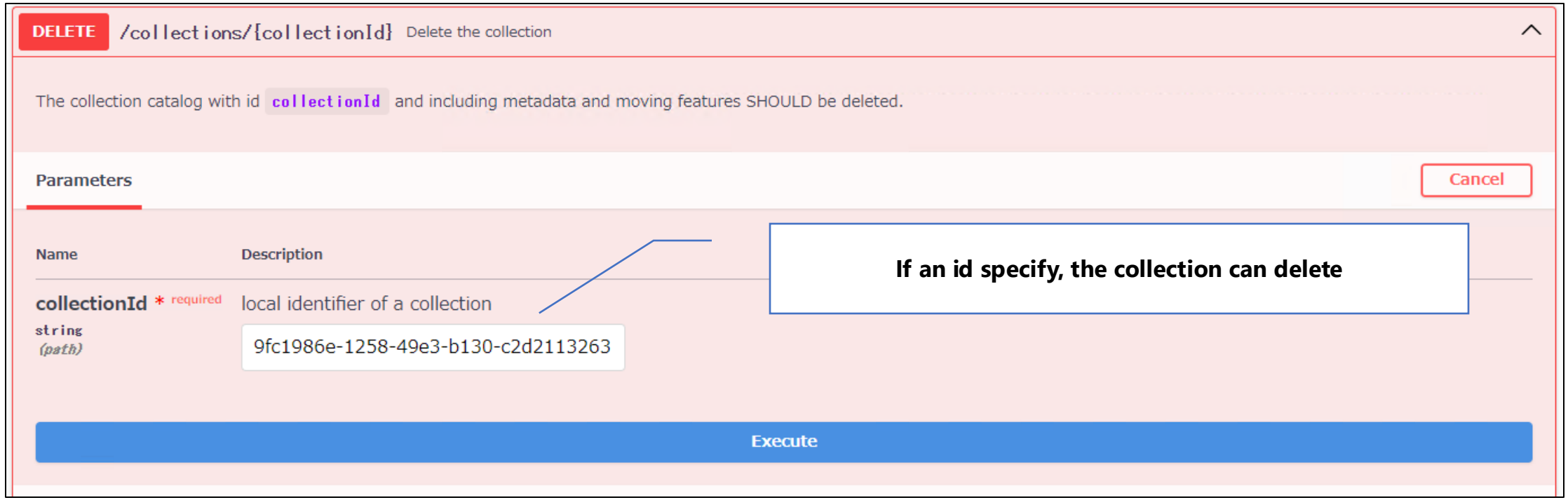
Input parameters when POST


```
{
  "collections": [
    {
      "title": "moving_feature_collection_sample1",
      "description": "FOSS4G2023 Seoul Workshop",
      "updateFrequency": 1000,
      "itemType": "movingfeature",
      "id": "9fc1986e-1258-49e3-b130-c2d211326319",
      "extent": {
        "spatial": {
          "bbox": [],
          "crs": "http://www.opengis.net/def/crs/OGC/1.3/CRS84"
        }
      },
      "temporal": {
        "interval": [],
        "trs": "http://www.opengis.net/def/uom/ISO-8601/0/Gregorian"
      },
      "links": [
        {
          "href": "http://localhost:8085/collections/9fc1986e-1258-49e3-b130-c2d211326319",
          "rel": "self",
          "type": "application/json"
        }
      ]
    }
  ],
  "links": [
```

# APIs about MovingFeatureCollection

## DELETE /collections/{collectionId}

**Skip the DELETE request for the next demonstration**



# APIs about MovingFeatures

**POST** /collections/{collectionId}/items

No5

# APIs about MovingFeatures

## POST /collections/{collectionId}/items

No5

**Result**

Server response

| Code | Details |
|---|---|
| 201 | Response headers |

```
access-control-allow-origin: http://localhost:8085
connection: close
content-language: en-US
content-length: 0
content-type: application/json
date: Fri,17 Nov 2023 01:17:43 GMT
location: http://localhost:8085/collections/9fc1986e-1258-49e3-b130-c2d211326319/items/f7a5b599-389e-4fdf-8849-bae0221dbe0c
server: Werkzeug/2.3.6 Python/3.9.2
vary: Origin
x-powered-by: pygeoapi 0.14.dev0
```

mFeatureId

Responses

| Code | Description |
|---|---|
| 201 | Successful create a set of moving features or a moving feature into a specific collection. |

**mFeatureId : f7a5b599-389e-4fdf-8849-bae0221dbe0c**

# APIs about MovingFeatures

## POST /collections/{collectionId}/items

No5-2


### Parameters

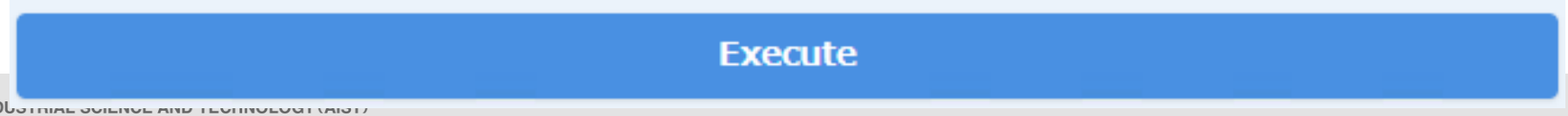
collectionId

9fc1986e-1258-49e3-b130-c2d211326319

Reference GitHub Page

Request body(application/json)

```
{
  "type": "Feature",
  "properties": {
    "name": "car1", ...
```



# APIs about MovingFeatures

## POST /collections/{collectionId}/items

No5-2

## Result

Server response

| Code | Details |
|---|---|
| 201 | Response headers |

```
access-control-allow-origin: http://localhost:8085
connection: close
content-language: en-US
content-length: 0
content-type: application/json
date: Fri,24 Nov 2023 04:21:25 GMT
location: http://localhost:8085/collections/f71c3c59-becd-4cb7-bd98-a93505e769c8/items/015b3091-2a77-4390-b473-2966b64a8cb8
server: Werkzeug/2.3.7 Python/3.9.2
vary: Origin
x-powered-by: pygeoapi 0.14.dev0
```

mFeatureId

Responses

| Code | Description |
|---|---|
| 201 | Successful create a set of moving features or a moving feature into a specific collection. |

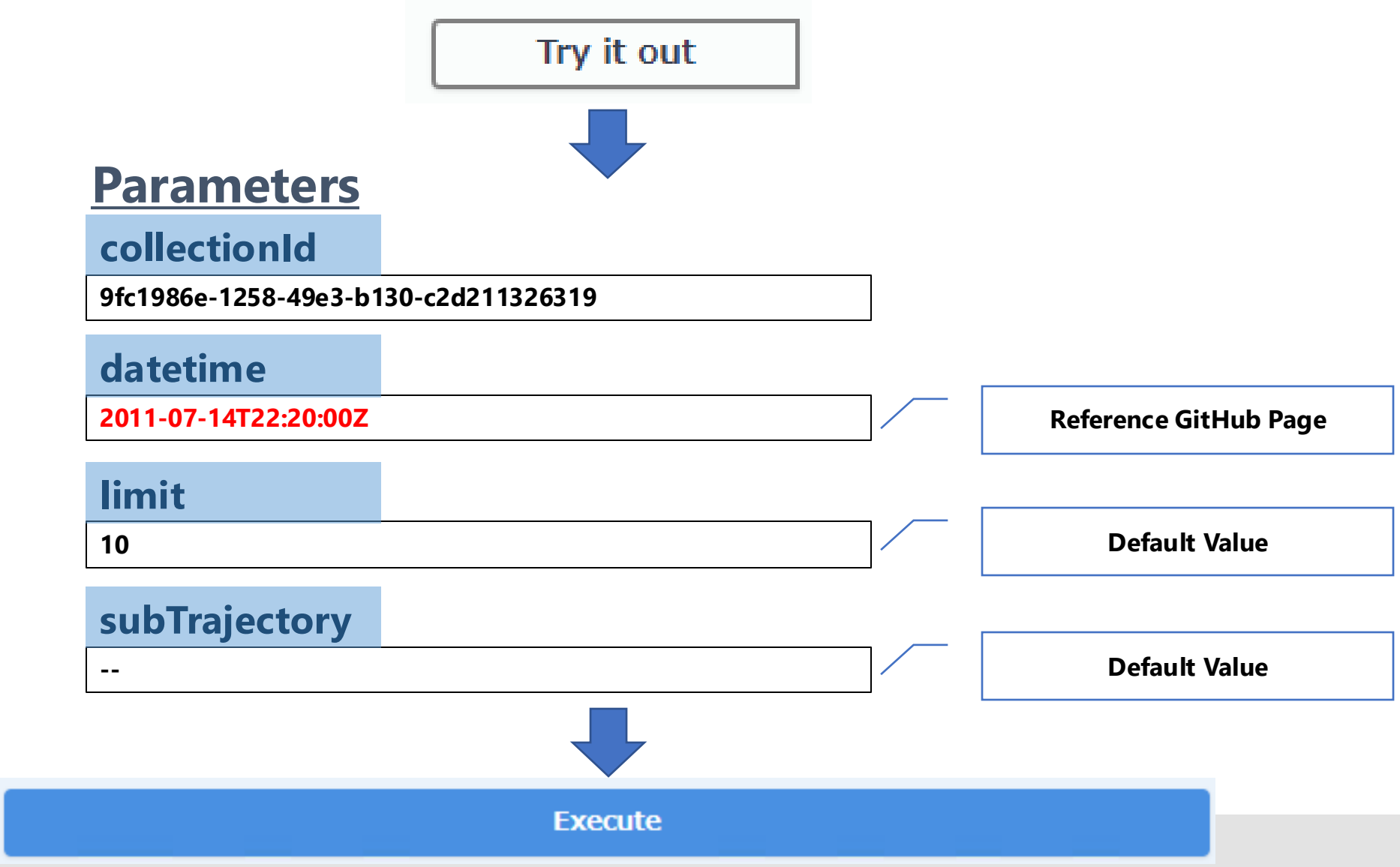
**mFeatureId :015b3091-2a77-4390-b473-2966b64a8cb8**

# APIs about MovingFeatures

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items

No6



# APIs about MovingFeatures

## GET /collections/{collectionId}/items

No6

## Result

Server response

Code | Details

200

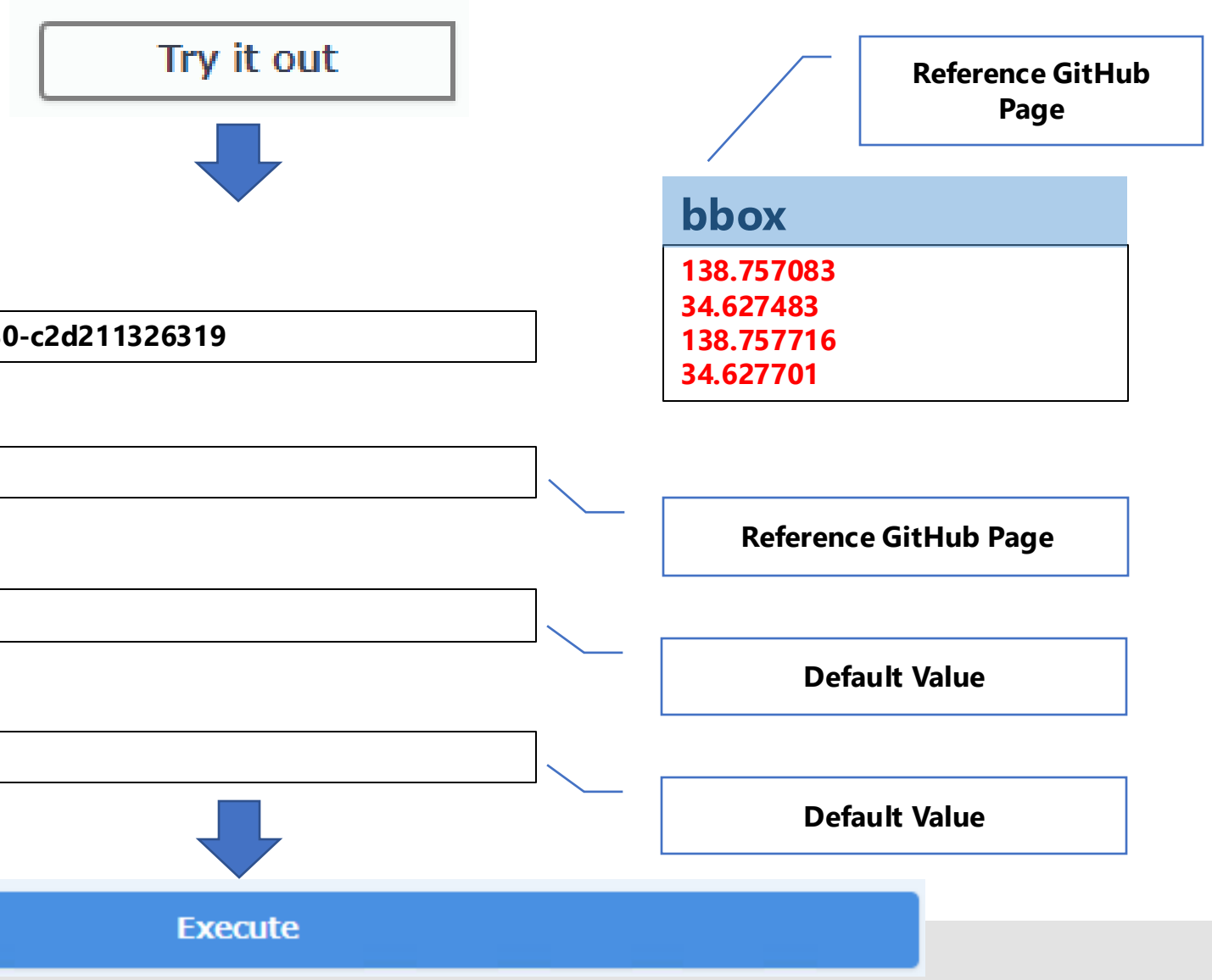
Response body


```
{
  "type": "FeatureCollection",
  "features": [
    {
      "id": "f7a5b599-389e-4fdf-8849-bae0221dbe0c",
      "crs": {
        "type": "Name",
        "properties": {
          "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
        }
      },
      "trs": {
        "type": "Link",
        "properties": {
          "href": "http://www.opengis.net/def/uom/ISO-8601/0/Gregorian",
          "type": "ogcdef"
        }
      },
      "type": "Feature",
      "properties": {
        "name": "car1",
        "state": "test1",
        "video": "http://.../example/video.mpeg"
      },
      "geometry": {
        "type": "LineString",
        "coordinates": [
          [
```

# APIs about MovingFeatures

## GET /collections/{collectionId}/items

No6-2



# APIs about MovingFeatures

## GET /collections/{collectionId}/items

**No6-2**

## Result

Server response

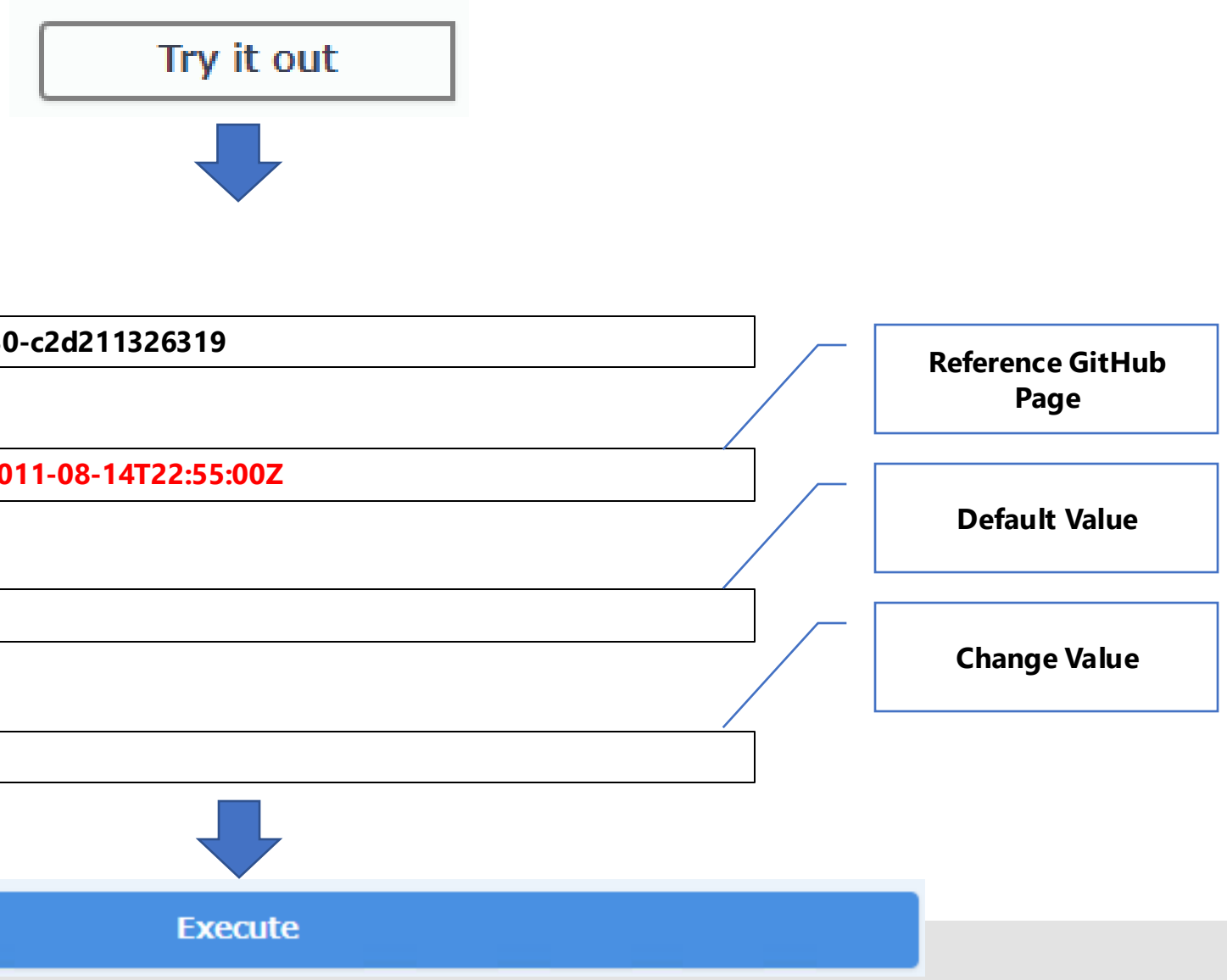
| Code | Details |
| --- | --- |
| 200 | Response body |


```
{
  "type": "FeatureCollection",
  "features": [
    {
      "id": "f7a5b599-389e-4fdf-8849-bae0221dbe0c",
      "crs": {
        "type": "Name",
        "properties": {
          "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
        }
      },
      "trs": {
        "type": "Link",
        "properties": {
          "href": "http://www.opengis.net/def/uom/ISO-8601/0/Gregorian",
          "type": "ogcdef"
        }
      },
      "type": "Feature",
      "properties": {
        "name": "car1",
        "state": "test1",
        "video": "http://.../example/video.mpeg"
      },
      "geometry": {
        "type": "LineString",
        "coordinates": [
          [
```

# APIs about MovingFeatures

## GET /collections/{collectionId}/items

No6-3



# APIs about MovingFeatures

## GET /collections/{collectionId}/items

No6-3

## Result

Server response

Code | Details

200

Response body

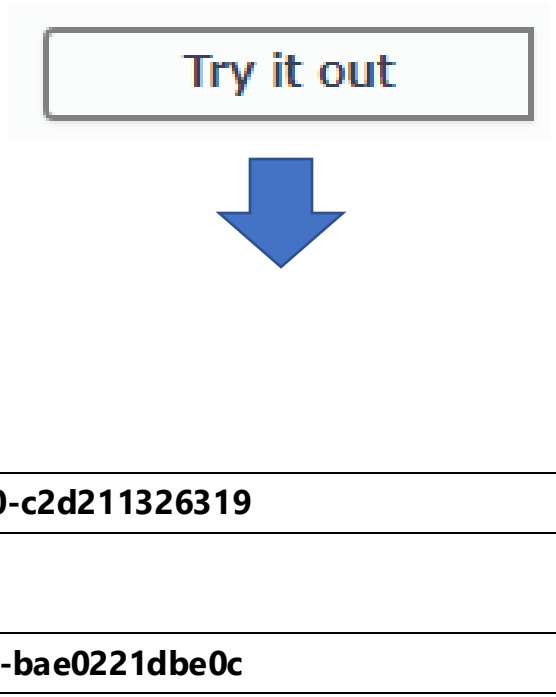

```
{
  "type": "FeatureCollection",
  "features": [
    {
      "id": "f7a5b599-389e-4fdf-8849-bae0221dbe0c",
      "crs": {
        "type": "Name",
        "properties": {
          "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
        }
      },
      "trs": {
        "type": "Link",
        "properties": {
          "href": "http://www.opengis.net/def/uom/ISO-8601/0/Gregorian",
          "type": "ogcdef"
        }
      },
      "type": "Feature",
      "properties": {
        "name": "car1",
        "state": "test1",
        "video": "http://.../example/video.mpeg"
      },
      "geometry": {
        "type": "LineString",
        "coordinates": [
          [
```

# APIs about MovingFeatures

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}

No7



# APIs about MovingFeatures

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}

**No7**

## Result

Server response

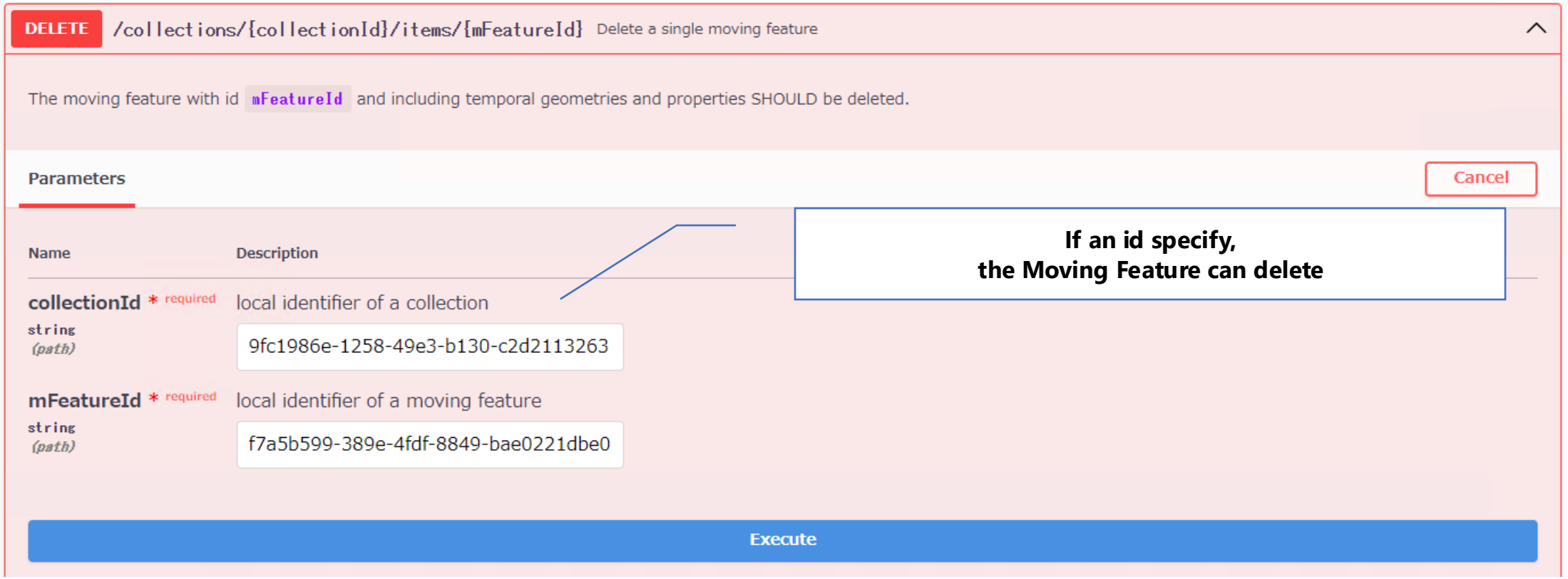
| Code | Details |
| --- | --- |
| 200 | Response body |


```
{
  "id": "f7a5b599-389e-4fdf-8849-bae0221dbe0c",
  "crs": {
    "type": "Name",
    "properties": {
      "name": "urn:ogc:def:crs:OGC:1.3:CRS84"
    }
  },
  "trs": {
    "type": "Link",
    "properties": {
      "href": "http://www.opengis.net/def/uom/ISO-8601/0/Gregorian",
      "type": "ogcdef"
    }
  },
  "type": "Feature",
  "properties": {
    "name": "car1",
    "state": "test1",
    "video": "http://.../example/video.mpeg"
  },
  "geometry": {
    "type": "LineString",
    "coordinates": [
      [
        139.757083,
        35.627701,
        0.5
```

# APIs about MovingFeatures

## DELETE /collections/{collectionId}/items/{mFeatureId}

**Skip the DELETE request for the next demonstration**



# APIs about TemporalGeometry

**POST** /collections/{collectionId}/items/{mFeatureId}/tgsequence

No9

# APIs about TemporalGeometry

**POST /collections/{collectionId}/items/{mFeatureId}/tgsequence**

**No9**

## Result

Server response

| Code | Details |
|---|---|
| 201 | Response headers |

```
access-control-allow-origin: http://localhost:8085
connection: close
content-language: en-US
content-length: 0
content-type: application/json
date: Fri,17 Nov 2023 01:47:16 GMT
location: http://localhost:8085/collections/9fc1986e-1258-49e3-b130-c2d211326319/items/f7a5b599-389e-4fdf-8849-bae0221dbe0c/tGeometries/8602827f-6927-4c3c-9057-12c386889774
server: Werkzeug/2.3.6 Python/3.9.2
vary: Origin
x-powered-by: pygeoapi 0.14.dev0
```

tGeometryId

Responses

| Code | Description | Links |
|---|---|---|
| 201 | Successful add more movement data into a specified moving feature. | *No links* |

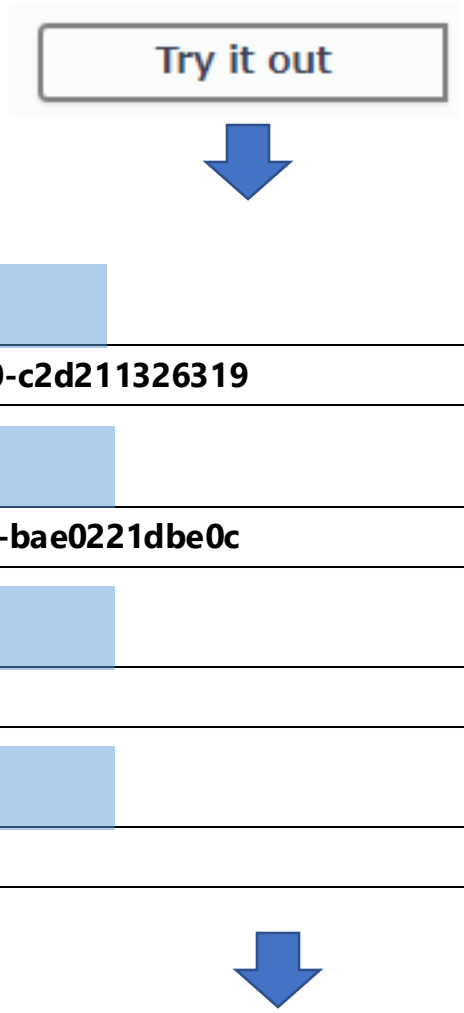
**tGeometryId : 8602827f-6927-4c3c-9057-12c386889774**

# APIs about TemporalGeometry

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}/tgsequence

No10



# APIs about TemporalGeometry

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}/tgsequence

**No10**

## Result

Server response

Code | Details

200

Response body


```
            35.627483,
            4
          ]
        ],
        "datetimes": [
          "2011-07-14T22:01:06-09Z",
          "2011-07-14T22:01:07-09Z",
          "2011-07-14T22:01:08-09Z",
          "2011-07-14T22:01:09-09Z",
          "2011-07-14T22:01:10-09Z"
        ],
        "interpolation": "Linear",
        "id": "8602827f-6927-4c3c-9057-12c386889774"
      }
    ],
    "crs": {
      "type": "Name",
      "properties": "urn:ogc:def:crs:OGC:1.3:CRS84"
    },
    "trs": {
      "type": "Name",
      "properties": "urn:ogc:data:time:iso8601"
    },
    "links": [
      {
        "href": "http://localhost:8085/collections/9fc1986e-1258-49e3-b130-c2d211326319/ite
        "rel": "self",
        "type": "application/json"
```

# APIs about TemporalGeometry

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}/tgsequence

No10-2

Try it out

### Parameters

**collectionId**

9fc1986e-1258-49e3-b130-c2d211326319

**mFeatureId**

f7a5b599-389e-4fdf-8849-bae0221dbe0c

**datetime**

2011-08-14T22:15:00Z/2011-08-14T23:15:00Z

**limit**

10

**subTrajectory**

true

**leaf,datetime**

empty

Execute

# APIs about TemporalGeometry

## GET /collections/{collectionId}/items/{mFeatureId}/tgsequence

**No10-2**

## Result

Server response

Code | Details

200

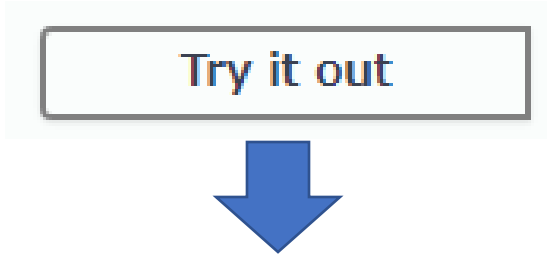
Response body


```
            35.627483,
            4
          ]
        ],
        "datetimes": [
          "2011-07-14T22:01:06-09Z",
          "2011-07-14T22:01:07-09Z",
          "2011-07-14T22:01:08-09Z",
          "2011-07-14T22:01:09-09Z",
          "2011-07-14T22:01:10-09Z"
        ],
        "interpolation": "Linear",
        "id": "8602827f-6927-4c3c-9057-12c386889774"
      }
    ],
    "crs": {
      "type": "Name",
      "properties": "urn:ogc:def:crs:OGC:1.3:CRS84"
    },
    "trs": {
      "type": "Name",
      "properties": "urn:ogc:data:time:iso8601"
    },
    "links": [
      {
        "href": "http://localhost:8085/collections/9fc1986e-1258-49e3-b130-c2d211326319/ite
        "rel": "self",
        "type": "application/json"
```

# APIs about TemporalGeometry

**POST** /collections/{collectionId}/items/{mFeatureId}/tproperties

No12



# APIs about TemporalGeometry

AIST
Create the Future, Collaborate Together

## POST /collections/{collectionId}/items/{mFeatureId}/tproperties

No12

## Result

Server response

| Code | Details |
|---|---|
| 201 | Response headers |

```
access-control-allow-origin: http://localhost:8085
connection: close
content-language: en-US
content-length: 0
content-type: application/json
date: Mon,20 Nov 2023 04:16:19 GMT
location: http://localhost:8085/collections/8f7c2c94-82db-4cad-b3f2-27c13ca1e5c5/items/1be529cd-fb42-416f-ab2e-3ed1f8e26c6d/tProperties/labels
server: Werkzeug/2.3.7 Python/3.9.2
vary: Origin
x-powered-by: pygeoapi 0.14.dev0
```

Responses

| Code | Description |
|---|---|

# APIs about TemporalProperties

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}/tproperties

No13

Try it out

### Parameters

**collectionId**

9fc1986e-1258-49e3-b130-c2d211326319

**mFeatureId**

f7a5b599-389e-4fdf-8849-bae0221dbe0c

**datetime**

2011-08-14T22:15:00Z/2011-08-14T22:55:00Z

**limit**

10

**subTemporalValue**

true

Execute

# APIs about TemporalProperty

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}/tproperties

No13

## Result

Server response

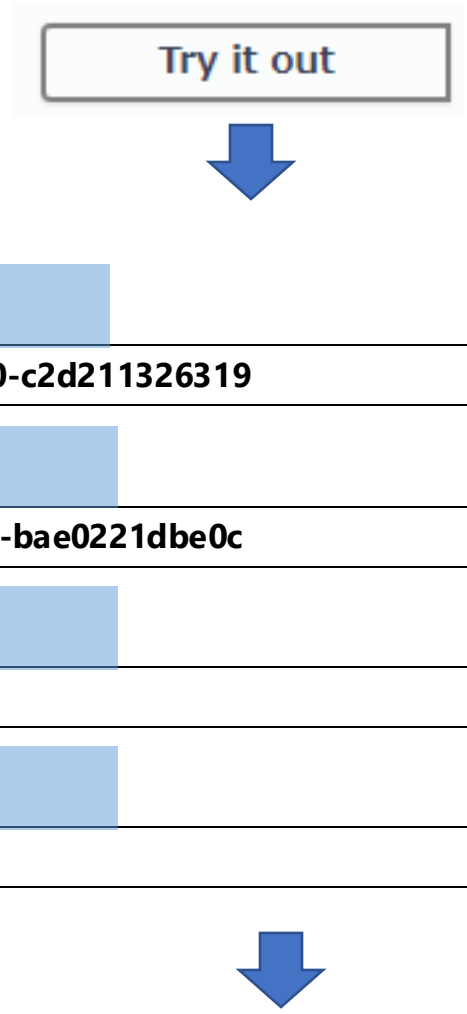
| Code | Details |
| --- | --- |
| 200 | Response body |


```
{
  "temporalProperties": [
    {
      "type": "Image",
      "name": "camera"
    },
    {
      "form": "MQS",
      "type": "Measure",
      "name": "discharge"
    },
    {
      "type": "Text",
      "name": "labels"
    },
    {
      "form": "http://www.qudt.org/qudt/owl/1.0.0/quantity/Length",
      "type": "Measure",
      "description": "description1",
      "name": "length"
    }
  ],
  "links": [
    {
      "href": "http://localhost:8085/collections/9fc1986e-1258-49e3-b130-c2c
      "rel": "self",
```

# APIs about TemporalProperty

GET /collections/{collectionId}/items/{mFeatureId}/tproperties/{tPropertyName}

No14



# APIs about TemporalProperty

AIST
Create the Future, Collaborate Together

## GET /collections/{collectionId}/items/{mFeatureId}/tproperties/{tPropertyName}

**No14**

## Result

Server response

| Code | Details |
|---|---|

200

Response body


```
{
  "temporalProperties": [
    {
      "values": [
        1,
        2.4,
        1
      ],
      "datetimes": [
        "2011-07-14T22:01:01.45-09Z",
        "2011-07-14T23:01:01.45-09Z",
        "2011-07-15T00:01:01.45-09Z"
      ],
      "interpolation": "Linear"
    }
  ],
  "links": [
    {
      "href": "http://localhost:8085/collections/8f7c2c94-82db-4cad-b3f2-27c13ca1e5c5/items/1be529cd-fb42-416f-ab2e-3ed1f8e26c6d/tProperties/length?offset=0&limit=10",
      "rel": "self",
      "type": "application/json"
    }
  ],
  "timeStamp": "2023-11-20T04:17:45.377817Z",
  "numberMatched": 1,
  "numberReturned": 1
}
```

# APIs about TemporalProperty

**POST** /collections/{collectionId}/items/{mFeatureId}/tproperties/{tPropertyName}

No15


Reference GitHub Page

# APIs about TemporalProperty

**POST** /collections/{collectionId}/items/{mFeatureId}/tproperties/{tPropertyName}

**No15**

## Result

Server response

Code | Details

201

Response headers

```
access-control-allow-origin: http://localhost:8085
connection: close
content-language: en-US
content-length: 0
content-type: application/json
date: Mon,20 Nov 2023 04:19:15 GMT
location: http://localhost:8085/collections/8f7c2c94-82db-4cad-b3f2-27c13ca1e5c5/items/1be529cd-fb42-416f-ab2e-3ed1f8e26c6d/tProperties/length/pvalue/0a59f0f1-2281-4133-811b-339b4863ce5c
server: Werkzeug/2.3.7 Python/3.9.2
vary: Origin
x-powered-by: pygeoapi 0.14.dev0
```

Responses

**GET** /collections/{collectionId}/items/{mFeatureId}/tproperties/{tPropertyName}

Server response

Code | Details

200

Response body


```
{
  "temporalProperties": [
    {
      "values": [
        0,
        20,
        50
      ],
      "datetimes": [
        "2011-07-15T08:00:00-09Z",
        "2011-07-15T08:00:01-09Z",
        "2011-07-15T08:00:02-09Z"
      ],
      "interpolation": "Linear"
    },
    {
      "values": [
        1,
        2.4,
        1
      ],
      "datetimes": [
        "2011-07-14T22:01:01.45-09Z",
        "2011-07-14T23:01:01.45-09Z",
        "2011-07-15T00:01:01.45-09Z"
      ],
      "interpolation": "Linear"
    }
```


confirm additions

# APIs about TemporalProperty

AIST
Create the Future, Collaborate Together

## DELETE /collections/{collectionId}/items/{mFeatureId}/tproperties/{tPropertyName}/{tValueId}

**No16**

Parameters | Cancel

| Name | Description |
|---|---|
| **collectionId** * required<br>`string`<br>*(path)* | local identifier of a collection<br>cb76b757-9c83-439b-baa9-e67c1cf12d74 |
| **mFeatureId** * required<br>`string`<br>*(path)* | local identifier of a moving feature<br>71426252-f252-44cc-9d13-25eb1b93c31b |
| **tPropertyName** * required<br>`string`<br>*(path)* | local identifier of a temporal property<br>label |
| **tValueId** * required<br>`string`<br>*(path)* | local identifier of a temporal primitive value<br>d3b5b6a4-2415-4681-b5c1-dbdb397e6bc9 |

# APIs about TemporalGeometryQuery

AIST
Create the Future, Collaborate Together

GET /collections/{collectionId}/items/{mFeatureId}/tgsequence/{tGeometryId}/acceleration

No17

Try it out

## Parameters

**collectionId**

cb76b757-9c83-439b-baa9-e67c1cf12d74

**mFeatureId**

71426252-f252-44cc-9d13-25eb1b93c31b

**mFeatureId**

0a7ac9ae-bed8-4c8b-9290-251e6ce21db9

**datetime**

2011-08-14T22:15:00Z/2011-08-14T23:55:00Z

**leaf**

**subTemporalValue**

--

Execute

# APIs about TemporalGeometryQuery

AIST
Create the Future, Collaborate Together

GET /collections/{collectionId}/items/{mFeatureId}/tgsequence/{tGeometryId}/acceleration

No17

## Result

| Code | Details |
| --- | --- |
| 200 | Response body |


```
{
  "form": "MTS",
  "name": "acceleration",
  "type": "TReal",
  "valueSequence": [
    {
      "datetimes": [
        "2011-08-14T22:20:00.000000Z",
        "2011-08-14T22:30:00.000000Z",
        "2011-08-14T22:40:00.000000Z",
        "2011-08-14T23:00:00.000000Z"
      ],
      "interpolation": "Step",
      "values": [
        0,
        0,
        0,
        0
      ]
    }
  ]
}
```

# APIs about TemporalGeometryQuery

AIST
Create the Future, Collaborate Together

GET /collections/{collectionId}/items/{mFeatureId}/tgsequence/{tGeometryId}/distance

No18

Try it out

## Parameters

**collectionId**

cb76b757-9c83-439b-baa9-e67c1cf12d74

**mFeatureId**

71426252-f252-44cc-9d13-25eb1b93c31b

**mFeatureId**

0a7ac9ae-bed8-4c8b-9290-251e6ce21db9

**datetime**

2010-01-01T00:00:00Z/..

**leaf**

2011-08-14T22:30:00Z

**subTemporalValue**

--

Execute

# APIs about TemporalGeometryQuery

AIST
Create the Future, Collaborate Together

**GET /collections/{collectionId}/items/{mFeatureId}/tgsequence/{tGeometryId}/distance**

**No18**

## Result

| Code | Details |
|---|---|
| 200 | Response body |


```
{
  "form": "MTR",
  "name": "distance",
  "type": "TReal",
  "valueSequence": [
    {
      "datetimes": [
        "2011-08-14T22:10:00.000000Z",
        "2011-08-14T22:20:00.000000Z",
        "2011-08-14T22:30:00.000000Z",
        "2011-08-14T22:40:00.000000Z",
        "2011-08-14T23:00:00.000000Z"
      ],
      "interpolation": "Linear",
      "values": [
        0,
        1.500000033285333,
        3.500000039411582,
        3.500133005572702,
        3.500263366609814
      ]
    }
  ]
}
```

# APIs about TemporalGeometryQuery

AIST
Create the Future, Collaborate Together

GET /collections/{collectionId}/items/{mFeatureId}/tgsequence/{tGeometryId}/velocity

No19

Try it out

## Parameters

**collectionId**

cb76b757-9c83-439b-baa9-e67c1cf12d74

**mFeatureId**

71426252-f252-44cc-9d13-25eb1b93c31b

**mFeatureId**

0a7ac9ae-bed8-4c8b-9290-251e6ce21db9

**datetime**

2011-08-14T22:15:00Z/2011-08-14T23:55:00Z

**leaf**

**subTemporalValue**

true

Execute

# APIs about TemporalGeometryQuery

AIST
Create the Future, Collaborate Together

GET /collections/{collectionId}/items/{mFeatureId}/tgsequence/{tGeometryId}/velocity

No19

## Result

| Code | Details |
| --- | --- |
| 200 | Response body |


```
{
  "form": "MTS",
  "name": "velocity",
  "type": "TReal",
  "valueSequence": [
    {
      "datetimes": [
        "2011-08-14T22:10:00.000000Z",
        "2011-08-14T22:20:00.000000Z",
        "2011-08-14T22:30:00.000000Z",
        "2011-08-14T22:40:00.000000Z",
        "2011-08-14T23:00:00.000000Z"
      ],
      "interpolation": "Step",
      "values": [
        0.002500000055476,
        0.003333333343544,
        2.21610269e-7,
        1.08634198e-7,
        1.08634198e-7
      ]
    }
  ]
}
```

# Thank you