

cyclomedia

Atlas Oblique Recording Service



Versie 20220517

17-05-2022

©2022 Cyclomedia Technology

Index

- INDEX 2**
- 1. INTRODUCTION..... 3**
 - 1.1 PURPOSE..... 3
 - 1.2 AUDIENCE..... 3
- 2. REQUESTS 3**
 - 2.1 AUTHENTICATION 3
 - 2.2 OPERATIONS..... 3
 - 2.3 RESPONSE FORMAT 4
- 3. OPERATIONS 4**
 - 3.1 COMMON 4
 - 3.2 GETCAPABILITIES..... 5
 - 3.3 DESCRIBEFEATURETYPE..... 6
 - 3.4 GETFEATURE 7
- 4. RESPONSE..... 10**
- 5. IMAGERY..... 12**
 - 5.1 OBLIQUE IMAGERY 12
 - 5.2 OBLIQUE PHOTOGRAPHS..... 12
 - 5.3 NADIR PHOTOGRAPHS 13



1. Introduction

1.1 Purpose

This document describes the Application Programmer's Interface (API) offered by the Atlas Oblique Recording service. The latest functionalities described in this document are implemented in 2017 July deployment of Oblique Recording service.

1.2 Audience

This document is intended for developers that want to use the Atlas Oblique Recording service.

2. Requests

2.1 Authentication

The HTTP Basic Authentication method is used to authenticate requests to the Oblique Recording service. A username and password can be obtained from CycloMedia.

2.2 Operations

The service is available from <https://atlasapi.cyclomedia.com/api/recording/wfs>. The HTTPS protocol is used to provide encrypted communication and secure identification of the web server.

The Oblique Recording service is a REST style service. It is a Web Feature Service (WFS) which supports the following operations:

- GetCapabilities, outputs metadata document describing the WFS service;
- DescribeFeatureType, outputs metadata document with the description of supported feature types;
- GetFeature, outputs the feature after retrieving it from the data source.

Both HTTP GET and HTTP POST requests are supported. The body of HTTP POST request is always in XML format.

2.3 Response format

The Oblique Recording service supports 2 response formats: JSON and XML. If the operation supports JSON and “application/json” is passed along, then JSON will be the output format, otherwise response format will default to XML.

- GetCapabilities, supports JSON and XML
- DescribeFeatureType, supports JSON and XML
- GetFeature, supports only XML

3. Operations

3.1 Common

These parameters can be used within all the supported operations.

Parameters

Attribute	Required	Description
service	Yes	Name of the service. Only valid value is WFS. Note: this value is case sensitive.
version	Yes	Version of the service. Only valid value is 1.1.0. Note: abbreviations like 1.1 are not valid.
request	Yes	Name of the operation. Valid values are: <ul style="list-style-type: none">• GetCapabilities• DescribeFeatureType• GetFeature Note: these values are case sensitive.
outputFormat	No	Output format of the response. Valid value is application/json. Returns JSON when application/json is supplied and if the operation supports JSON response, otherwise defaults to XML.

3.2 GetCapabilities

Outputs metadata document describing the WFS service.

Parameters

See 3.1 Common

Request format

The request has the following format:

GET

XML example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0  
&request=GetCapabilities
```

JSON example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0  
&request=GetCapabilities&outputFormat=application/json
```

POST

Post request are sent to URI <https://atlasapi.cyclomedia.com/api/recording/wfs> with one of the following request bodies:

XML example

```
<GetCapabilities  
  service="WFS"  
  version="1.1.0"  
  outputFormat="application/json"  
>
```

JSON example:

```
<GetCapabilities  
  service="WFS"  
  version="1.1.0"  
>
```

3.3 DescribeFeatureType

Outputs metadata document with the description of supported feature types.

Parameters

Attribute	Required	Description
typeName	No	Name of the feature type to describe. Only valid value is atlas:ObliqueImage. Note: if not passed atlas:ObliqueImage is used as default. The value is case sensitive.

Request format

The request has the following format:

GET

With typename example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0
&request=DescribeFeatureType
```

Without typename example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0
&request=DescribeFeatureType&typename=atlas:ObliqueImage
```

POST

Post request are sent to URI <https://atlasapi.cyclomedia.com/api/recording/wfs> with one of the following request bodies:

With typename example:

```
<DescribeFeatureType
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  service="WFS"
  version="1.1.0">
  <wfs:TypeName>atlas:ObliqueImage</wfs:TypeName>
</DescribeFeatureType>
```

Without typename example:

```
<DescribeFeatureType
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  service="WFS"
  version="1.1.0" />
```

3.4 GetFeature

Outputs the feature after retrieving it from the data source.

Parameters

Attribute	Required	Description
typeName	No	Name of the feature type to describe. Only valid value is atlas:ObliqueImage. Note: if not passed atlas:ObliqueImage is used as default. The value is case sensitive.
srsName	No	Specifies the coordinate system.
maxFeatures	No	Maximum number of features to return. Use maxFeatures to limit the results when featureId is unknown. If not supplied will default to 10000.
bbox	No	This attribute is the abbreviation of bounding box and it returns features within the specified coordinates of the bounding box. Only supported with GET requests.
featureId	Yes	ID of a specific feature. Limits the output to 1 result.
filter	No	To narrow down the result.

Use maxFeatures in combination with bbox or filter to limit the output. featureId always returns 0 or 1 result.

Request format

The request has the following format:

GET

FeatureID example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0&request=GetFeature&typename=atlas:ObliqueImage&srsname=EPSG:28992&featureid=<featureid>
```

BBOX example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0&request=GetFeature&typename=atlas:ObliqueImage&srsname=EPSG:28992&bbox=<bbox>
```

Filter example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0&request=GetFeature&typename=atlas:ObliqueImage&srsname=EPSG:28992&filter=<filter>
```

MaxFeatures example:

```
https://atlasapi.cyclomedia.com/api/recording/wfs?service=WFS&version=1.1.0&request=GetFeature&typename=atlas:ObliqueImage&srsname=EPSG:28992&bbox=<bbox>&maxfeatures=<maxfeatures>
```

POST

Post request are sent to URI <https://atlasapi.cyclomedia.com/api/recording/wfs> with one of the following request bodies:

FeatureID example

```
<GetFeature
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  service="WFS"
  version="1.1.0"
  maxFeatures="100">
  <wfs:Query typeName="atlas:ObliqueImage" srsName="EPSG:28992">
    <Filter>
      <FeatureId fid="<featureid>" />
    </Filter>
  </wfs:Query>
</GetFeature>
```

BBOX example:

```
<GetFeature
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  service="WFS"
  version="1.1.0">
  <wfs:Query typeName="atlas:ObliqueImage" srsName="EPSG:28992">
    <Filter>
      <BBOX>
        <gml:Envelope srsName='EPSG:28992'>
          <gml:lowerCorner>111111.1 222222.2</gml:lowerCorner>
          <gml:upperCorner>333333.3 444444.4</gml:upperCorner>
        </gml:Envelope>
      </BBOX>
    </Filter>
  </wfs:Query>
</GetFeature>
```

Filter example:

```
<GetFeature
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  service="WFS"
  version="1.1.0">
  <wfs:Query typeName="atlas:ObliqueImage" srsName="EPSG:28992">
    <Filter>
      <And>
        <BBOX>
          <gml:Envelope srsName='EPSG:28992'>
            <gml:lowerCorner>111111.1 222222.2</gml:lowerCorner>
            <gml:upperCorner>333333.3 444444.4</gml:upperCorner>
          </gml:Envelope>
        </BBOX>
        <PropertyIsGreaterThanOrEqualTo>
          <PropertyName>heading</PropertyName>
          <Literal>45</Literal>
        </PropertyIsGreaterThanOrEqualTo>
      </And>
    </Filter>
  </wfs:Query>
</GetFeature>
```



```

    </PropertyIsGreaterThanOrEqualTo>
    <PropertyIsLessThanOrEqualTo>
      <PropertyName>heading</PropertyName>
      <Literal>135</Literal>
    </PropertyIsLessThanOrEqualTo>
  </And>
</Filter>
</wfs:Query>
</GetFeature>

```

MaxFeatures example:

```

<GetFeature
  xmlns:wfs="http://www.opengis.net/wfs"
  xmlns="http://www.opengis.net/ogc"
  xmlns:gml="http://www.opengis.net/gml"
  service="WFS"
  version="1.1.0"
  maxFeatures="5">
  <wfs:Query typeName="atlas:ObliqueImage" srsName="EPSG:28992">
    <Filter>
      <And>
        <BBOX>
          <gml:Envelope srsName='EPSG:28992'>
            <gml:lowerCorner>111111.1 222222.2</gml:lowerCorner>
            <gml:upperCorner>333333.3 444444.4</gml:upperCorner>
          </gml:Envelope>
        </BBOX>
        <PropertyIsGreaterThanOrEqualTo>
          <PropertyName>heading</PropertyName>
          <Literal>45</Literal>
        </PropertyIsGreaterThanOrEqualTo>
        <PropertyIsLessThanOrEqualTo>
          <PropertyName>heading</PropertyName>
          <Literal>135</Literal>
        </PropertyIsLessThanOrEqualTo>
      </And>
    </Filter>
  </wfs:Query>
</GetFeature>

```

4. Response

Responses of a request can be either XML or JSON as described in section “2.3 Response format”. The attributes, attribute types and descriptions are listed below for reference.

Property name	Type	Description
imageld	string	Readable identification of the oblique recording.
year	int	Year in which the oblique recording is recorded.
recordedAt	dateTime	Timestamp and offset to UTC of the oblique recording.
recordingLocation	gml:Point	X and Y coordinates of the recording location within the given EPSG.
recordingHeight	float	Height of the recording within the given EPSG.
viewDirection	String	Direction of the view. Values can be <ul style="list-style-type: none"> • N: North • W: West • E: East • S: South
footprintCentre	gml:Point	X and Y coordinate of the center relative to the 4 corners of the footprint.
footprint	gml:Polygon	X, Y and Z value of the 4 corners of the footprint belonging to the oblique image.
expiredAt	dateTime	Date when the oblique recording is expired.
width	int	Width of the oblique image in pixels.
height	int	Height of the oblique image in pixels.
rotated	boolean	Flag indicates if the oblique image should be rotated 180 degrees so that the horizon is upper side of the oblique image.
heading	float	The horizontal view direction in degrees of the oblique image.
roll	float	Rotation around the optical axis in degrees of the oblique image.
pitch	float	Comparable with the tilt direction of the oblique image.
f	float	Focal length in pixels of the oblique image.
dU	float	X coordinate of the principle point in pixels of the oblique image.
dV	float	Y coordinate of the principle point in pixels of the oblique image.
k1	float	Radial lens distortion parameter.
k2	float	Radial lens distortion parameter.
p1	float	Tangential lens distortion parameter.
p2	float	Tangential lens distortion parameter.
delta	float	Astigmatism lens distortion parameter.
alpha	float	Astigmatism lens distortion parameter.
isAuthorized	boolean	Indicates whether the user is authorized to view this oblique recording.

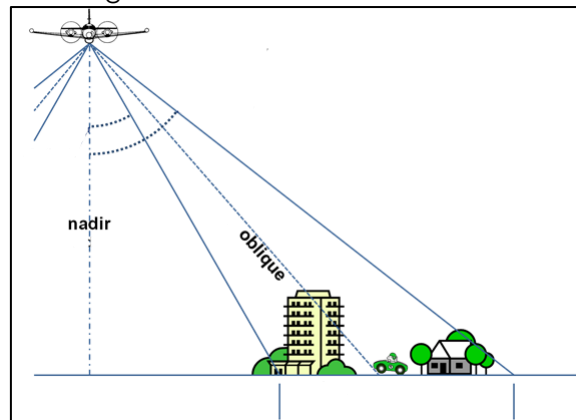
dataset	string	The name of the dataset where the oblique image is part of.
zoomlevels	int	The number of zoom levels supported by the oblique image.
published	boolean	Flag indicating if it is publicly visible.
groundPrecisionX	float	Precision on the ground for the X coordinate in meters.
groundPrecisionY	float	Precision on the ground for the Y coordinate in meters.
groundPrecisionZ	float	Precision on the ground for the Z coordinate in meters.
longitudePrecision	float	Precision of the longitude coordinate of the recording location in meters.
latitudePrecision	float	Precision of the latitude coordinate of the recording location in meters.
heightPrecision	float	Precision of the height of the recording location in meters.
headingPrecision	float	Precision of the horizontal view direction in degrees.
pitchPrecision	float	Precision of the pitch direction in degrees.
rollPrecision	float	Precision of the roll direction in degrees.

5. Imagery

Oblique Recording service returns metadata of oblique imagery. In this section oblique imagery is explained.

5.1 Oblique imagery

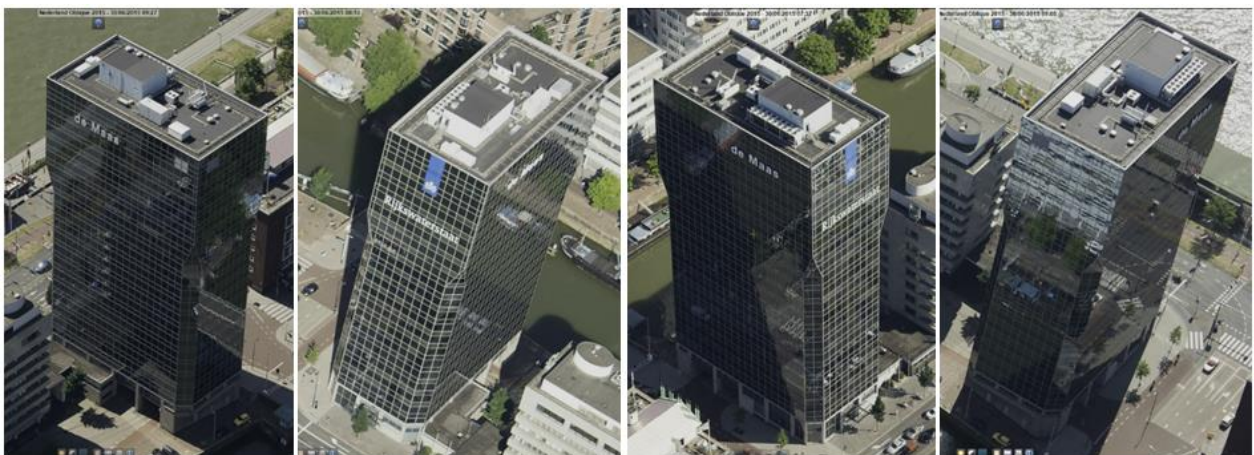
Oblique imagery of CycloMedia come in two forms: oblique and nadir. Both forms are taken at the same time at different angles from the earth's surface.



5.2 Oblique photographs

Oblique characteristic

- the optical axis of the camera has a sharp angle with the earth's surface.
- any object, e.g. buildings, on the photograph can be viewed from 4 sides: north, east, south and west.
- the default recording axis has an angle from -45 to -25 and from 25 to 45 degrees.



Example oblique from 4 sides: north, east, south and west.

5.3 Nadir photographs

Nadir characteristic

- the optical axis of the camera is perpendicular to the earth's surface.
- any object, e.g. buildings, on the photograph can be viewed from the top.
- the default recording axis has an angle from -20 to 0 and from 0 to 20 degrees.



Example nadir from the top.



**Visualize a
better world**