



METEO FRANCE
Toujours un temps d'avance

MET-GATE

Web Services Description

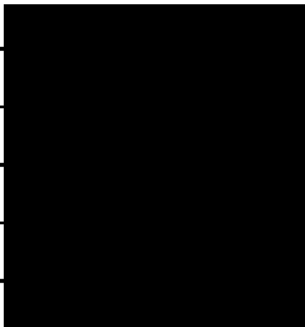
Version	Date	Author	Description
1.0	02/06/2015		Initial draft
1.0	02/06/2015		Initial draft
1.1	17/07/2015		update
1.1	17/07/2015		update
1.2	22/10/2015		Add Section 5
1.3	28/01/2016		Minor changes



Table of contents

1. Introduction	4
2. Web Feature Service	5
2.1. Feature types provided by the MET-GATE	5
2.1.1. SIGMET	5
2.1.2. Convection and Icing phenomena	5
2.2. WFS API	6
3. Web Coverage Service Application Profile - MetOcean	11
3.1. Coverages provided by the MET-GATE.....	11
3.2. WCS Application Profile – MetOcean API	12
4. Subscription Service.....	15
4.1. Subscription API	15
4.2. Automated polling configuration	21
5. Swim Mater Class Services.....	23
5.1. Airport MET forecast service (CHECK)	23
5.2. Airport MET observation service (CHECK)	23
5.3. MET gridded forecast service	24
5.4. MET Hazard En Route forecast service	24
5.5. MET Hazard En Route observation service	25
5.6. METAR service.....	26
5.7. TAF service	27
5.8. SIGMET service.....	27
References	29



List of tables

Table 1 WFS Operations 7

Table 2 WCS Operations..... 13

Table 3 Query Parameters GetCapabilities 13

Table 4 Query Parameters DescribeCoverage 14

Table 5 Query Parameters GetCoverage 14



1. Introduction

This document aims to describe and specify the available services provided by Meteo France through the MET-GATE prototype.

All available services requires authentication to be accessed. The method chosen is based on **basic access authentication** which is a method for an HTTP user agent to provide a user name and password when making a request. HTTP Basic authentication does not require cookies, session identifier and login pages. In order to request access to the available services, please send an email to **projet.sesar-dev@meteo.fr**

The current available services for ATM users are:

- Web Feature Service (WFS)
- Web Coverage Service (WCS) Application Profile MetOcean
- Subscription Service

Next sections will describe the capabilities of each service, focusing mostly on the subscription service since the WFS and WCS are standard services.



2. Web Feature Service

The WFS specification defines a set of interfaces for describing data manipulation operations of geographic features. The whole specification can be found in [1]

The MET-GATE WFS is based on the open source solution GeoServer [2]. The WFS Web API provided by GeoServer implements a REST architecture.

As the WFS is a standardized service, next sections will not focus on describing precisely the service API. Nevertheless some examples of usage are provided.

2.1. Feature types provided by the MET-GATE

2.1.1. SIGMET

SIGMET are supplied in IWXXM 1.0 format.

2.1.2. Convection and Icing phenomena

As of today the provided convection information comes from ASPOC3D and RDT products of Meteo France, and icing information is produced by DWD. Convection and Icing are supplied in a AIRM-compliant format. They are described through a WeatherCondition as shown in Figure 1 and Figure 2 below.

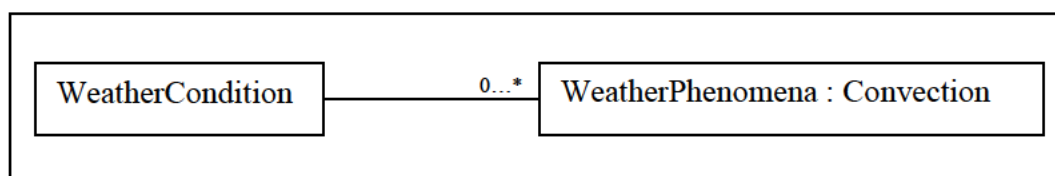


Figure 1 : schematic representation of convection phenomena in the MET-GATE

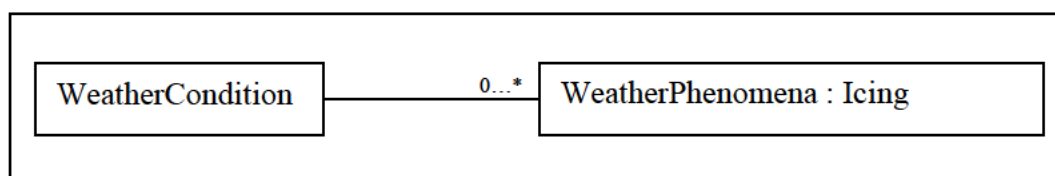


Figure 2: schematic representation of icing phenomena in the MET-GATE

The attributes of Weather Condition are:

- lastRevision : i.e. issueTime
- obsOrFcstTime : time of the physical phenomenon. If lastRevision is equal to obsOrFcstTime, this is an observation report
- list of WeatherPhenomena



A WeatherPhenomenon describes either a convection cell or an icing cell:

- WeatherPhenomena : Convection
 - Identifier : the cell identifier
 - directionToward : cell movement direction
 - speed : cell movement speed
 - extent : cell geographical extent defined by a polygon
 - intensity : enumerate value (NONE, LIGHT, MODERATE, MODERATE_SEVERE and SEVERE)
 - base: base of the cell. This will always be 0
 - top: top of the cell in FL
 - convectionCellTopVariation : information about variation of the top of the cell. Enumerate value (UNKNOWN, CONSTANT, DECAYING and GROWING.)
 - phaseType (BIRTH, MATURE and GROWING)
 - convectionType (ASPOC 3D or DRT)
- WeatherPhenomena : Icing
 - extent : cell geographical extent defined by a polygon
 - intensity : enumerate value (NONE, LIGHT, MODERATE and SEVERE)
 - base: base of the cell. This will always be 0
 - top : top of the cell in FL

2.2. WFS API

Endpoint

- <http://claudius.meteo.fr>

Resource

- [/geoserver/metgate/wfs/](#)

Supported methods

- GET requests for retrieving metadata about the wfs service in order to know how to use it and feature types of MET Information.
- POST requests for creating complex using filters. All GET requests are their equivalent using POST.

WFS Operations

The operations supported by the WFS are describe in Table 1

Operation	Description
GetCapabilities	Generates a metadata document describing a WFS service provided by the server as well as valid WFS operations and parameters
DescribeFeatureType	Returns a description of feature types supported by a WFS service
GetFeature	Returns a selection of features from a data source including geometry and attribute values
LockFeature	Prevents a feature from being edited through a persistent feature



	lock
Transaction	Edits existing feature types by creating, updating, and deleting

Table 1 WFS Operations

GetCapabilities

The “GetCapabilities” operation requests the list of the available operations and services supported by the server. Let’s see an example using GET request for MET-GATE WFS.

- `http://claudius.meteo.fr/geoserver/metgate/wfs?`
 `service=WFS`
 `&version=2.0.0`
 `&request=GetCapabilities`

DescribeFeatureType

The “DescribeFeatureType” provides information about the available feature types in the service.

- `http://claudius.meteo.fr/geoserver/metgate/wfs?`
 `service=WFS`
 `&version=2.0.0`
 `&request=DescribeFeatureType`
 `&typeName=airm`

GetFeature

The “GetFeature” operation returns a selection of features from the data source.

- `http://claudius.meteo.fr/geoserver/metgate/wfs?`
 `service=WFS`
 `&version=2.0.0`
 `&request=GetFeature`
 `&typeName=airm`

The previous request example, retrieves the whole set of AIRM features. In many cases (it depends on the volume information) can take forever to load, and has been known to crash browsers and so it is generally to be avoided (a lot of information to be loaded in the browser).

In order to limit the number of AIRM features, the parameter count can be used.

- `http://claudius.meteo.fr/geoserver/metgate/wfs?`
 `service=WFS`
 `&version=2.0.0`
 `&request=GetFeature`



&typeName=airm
&count=50
&outputFormat=GML2

It is important to notice that some of the query parameters are different between the WFS versions. For example, the parameter **count** must be replaced by **maxFeature** if using version **1.0.0** [1].

Others parameters can be added to customize a GetFeature request. Some examples of parameters are given below.

PropertyName

In order to recover only elements your are interesting in, it will be necessary to add a new parameter as follow: “propertyName=element[, element]” to the URL, where element is one of the elements described by the DescribeFeatureType operation.

For example, to recover AIRM features with only the elements “obsorfcsttime” and “intensity” the URL should look like:

- [http://claudius.meteo.fr/geoserver/metgate/wfs?
service=WFS
&version=2.0.0
&request=GetFeature
&typeName=airm
&propertyName=obsorfcsttime,intensity
&count=50
&outputFormat=GML2](http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=airm&propertyName=obsorfcsttime,intensity&count=50&outputFormat=GML2)

OrderBy

To sort features by alphabetical order, the “sortBy=element[+A|+D]” can be added to the URL. “A” means ascendant order and on the contrary “D” means descendant order .

- [http://claudius.meteo.fr/geoserver/metgate/wfs?
service=WFS
&version=2.0.0
&request=GetFeature
&typeName=airm
&sortBy=obsorfcsttime+D
&count=50
&outputFormat=GML2](http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=airm&sortBy=obsorfcsttime+D&count=50&outputFormat=GML2)

Filter query

More complex queries can be performed using the filter query parameter.



The next **POST request** gets all the AIRM features between 2015-06-01T10:00:00 and 2015-06-01T12:00:00 inclusive

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0" xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML2">
  <Query typeNames="airm">
    <fes:Filter>
      <fes:PropertyIsBetween>
        <fes:ValueReference>obsorfcsttime</fes:ValueReference>
        <fes:LowerBoundary>
          <fes:Literal>2015-06-01T10:00:00</fes:Literal>
        </fes:LowerBoundary>
        <fes:UpperBoundary>
          <fes:Literal>2015-06-01T12:00:00</fes:Literal>
        </fes:UpperBoundary>
      </fes:PropertyIsBetween>
    </fes:Filter>
  </Query>
</GetFeature>
```

Another **POST request** requesting AIRM data by using a bounding box as well as entitytype property.

```
<GetFeature
  service="WFS"
  version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:gml="http://www.opengis.net/gml/3.2"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML2">
  <Query typeNames="airm">
    <fes:Filter>
      <fes:And>
        <fes:BBOX>
          <fes:ValueReference>extent</fes:ValueReference>
          <gml:Envelope srsName="urn:ogc:def:crs:EPSG::4326">
            <gml:lowerCorner>40 -7</gml:lowerCorner>
            <gml:upperCorner>52 12</gml:upperCorner>
          </gml:Envelope>
        </fes:BBOX>
        <fes:PropertyIsEqualTo>
          <fes:ValueReference>entitytype</fes:ValueReference>
          <fes:Literal>Convection</fes:Literal>
        </fes:PropertyIsEqualTo>
      </fes:And>
    </fes:Filter>
  </Query>
</GetFeature>
```



Further documentation about the whole service capabilities can be found in [3]



3. Web Coverage Service Application Profile - MetOcean

The WCS Application Profile – MetOcean is an application profile of WCS which bundles and extends WCS functionally for serving coverage in the MetOcean domain, like climate and weather simulation, aviation, oceanography, etc [4]. It aims to access 4-D x/y/z/t coverages, called Weather Cubes.

The WCS Application Profile – MetOcean specification [5] is still under development and therefore the MET-GATE implementation could change according to new outputs of such specification.

3.1. Coverages provided by the MET-GATE

As of today, the MET-GATE is capable to provide AROME, CAT, CONVECTION_INDEX_GRID and ICING_GRID models. Next sections provide further details about them.

3.1.1. AROME

AROME is the high-resolution Meteo France's model. The geographical area covered by the model is the area defined by (38N, 9W ; 55N, 14E) with a resolution of 0.025° (~2.5km) . AROME provides forecast up to 36h with a 1h time step. It is updated every 6 hours

There is one coverage per physical parameter:

- Relative humidity
- Geopotential
- Reflectivity
- Temperature
- Wind speed
- Wind direction

The list of available vertical levels (FL) is [50, 65, 80, 100, 120, 140, 160, 180, 210, 240, 270, 300, 320, 340, 360, 390, 410, 450, 530] for all coverages except for the Reflectivity for which it is [50, 65, 80, 100, 120, 140, 160, 180, 210, 240, 270, 300, 340, 390].

3.1.2. CAT

Clear-Air-Turbulence information is not yet available. It will be provided by the UK Met Office.

The model features are:

- Coverage: (72N, 42E ; 20N, 32W)
- Resolution: 0.1° (~10km)
- FL : 240, 300, 340, 390, 450
- Forecast up to 30 hours ahead (in steps of 3 hours)



- Updated every 6 hours

The severity index will have 3 values: 1 for light, 2 for moderate and 3 for severe turbulence.

3.1.3. CONVECTION_INDEX_GRID

The convection index grids are based on AROME model. They have the same features than AROME, except the resolution which is 0.5° (~5km). They contain a convection severity index going from 0 to 4.

3.1.4. ICING_GRID

The icing grids are produced by DWD, the German weather national service.

The model features are:

- Coverage: (70N, 50E ; 30N, 40W)
- Resolution: 0.05° (~5km)
- FL : 20, 25, 30, 50, 65, 100, 140, 180, 240
- Forecasts up to 24 hours ahead (in steps of 3 hours)
- Updated every 6 hours

There are 3 icing severity index: 1=light,2=moderate,3=severe.

3.2. WCS Application Profile – MetOcean API

Endpoint

- <http://claudius.meteo.fr>

Resource

- [/geoserver/metgate/wcs/](#)

Supported methods

- GET queries for retrieving metadata about the service in order to know how to use it and coverages of MET Information.

Query Parameters

service [WCS]

The service name - determined WFS

version [2.0]

The service version - determined value 2.0

request [GetCapabilities / GetCapabilitiesEnsemble / DescribeCoverage / DescribeEnsemble GetCoverage]

The operation name to be requested to the service



coverageId [optional, string]

The unique id of a coverage

ensembleId[optional, string]

The unique id of an ensemble

subset [optional, time / height / latitude / longitude]

The sub selection coordinates to trim or slice a coverage. The syntax accepted by the subset values is as follow:

- Slice by time (minValue, maxValue)
- Trim by time (value).

The same applies to height, latitude and longitude.

outputFormat [optional, nc / ncz]

The data output format

Operations

Operation	Description
GetCapabilities	Generates a metadata document describing a WCS service provided by the server as well as valid WCS operations and parameters
DescribeCoverage	Returns a description of a coverage supported by the WCS service
GetCapabilitiesEnsemble	Generates a metadata document describing a WCS service provided by the server as well as valid WCS operations and parameters
DescribeEnsemble	Returns a description of an ensembles supported by the WCS service
GetCoverage	Returns a coverage in a well known format. The current implementation only supports NetCDF and compressed NetCDF

Table 2 WCS Operations

GetCapabilities

Table 3 shows the required query parameters when requesting a GetCapabilities operation.

Parameter	Required
service	yes
version	yes
request	yes

Table 3 Query Parameters GetCapabilities

Example

- <http://claudius.meteo.fr/geoserver/metgate/wcs/?service=WCS&version=2.0&request=GetCapabilities>



DescribeCoverage

Table 4 shows the required query parameters when requesting a DescribeCoverage operation.

Parameter	Required
service	yes
version	yes
request	yes
coverageId	yes

Table 4 Query Parameters DescribeCoverage

Example

- http://claudius.meteo.fr/geoserver/metgate/wcs/?service=WCS&version=2.0&request=DescribeCoverage&coverageId=GP_GDS0_ISBL_20150529000000

GetCoverage

Table 5 shows the required and optional query parameters when requesting a GetCoverage operation

Parameter	Required
service	yes
version	yes
request	yes
coverageId	yes
subset	no
outputFormat	no

Table 5 Query Parameters GetCoverage

Example

- http://claudius.meteo.fr/geoserver/metgate/wcs/?service=WCS&version=2.0&request=GetCoverage&coverageId=GP_GDS0_ISBL_20150721000000&outputFormat=ncz
- [http://claudius.meteo.fr/geoserver/metgate/wcs/?service=WCS&version=2.0&request=GetCoverage&coverageId=GP_GDS0_ISBL_20150721000000&subset=latitude\(49.125,50.925\)&subset=time\(1,4\)](http://claudius.meteo.fr/geoserver/metgate/wcs/?service=WCS&version=2.0&request=GetCoverage&coverageId=GP_GDS0_ISBL_20150721000000&subset=latitude(49.125,50.925)&subset=time(1,4))



4. Subscription Service

The subscription service allows subscribers to manage their own subscriptions in order to automatically retrieve the wished MET information provided by the MET-GATE.

The MET-GATE provides two ways of publishing information:

- **Automated polling:** every time a subscription is executed, metadata about the requested MET-Information is sent to a specific broker that an ATM program can use to retrieve the MET information automatically (no human intervention).
- **Email notification:** every time a subscription is executed, the system sends an email notification to an ATM client email, specifying an URL where the MET information can be downloaded.

Also, the MET-GATE provides an option that allows the ATM clients decide how the information is supplied to them:

- **On event:** MET Information is supplied to ATM client as soon as new available data arrives into the system
- **Schedule:** MET Information is supplied at regular intervals defined by the clients.

Next section specifies the subscription API to show how to configure a subscription by an ATM client.

4.1. Subscription API

Endpoint

- <http://claudius.meteo.fr>

Resource

- `/wcube-dataservice-rest-ejb/api/{apiVersion}/atmusers/{atmUserId}/subscriptions`

Path parameters

apiVersion

The version of the API (available versions: 1.0)

atmuserId

The unique id of an ATM user

Supported methods

- GET queries for a list of subscriptions associated to an ATM user.
- POST creates a new subscription and will provide the created subscription in the response body.
- PUT updates an existing subscription with the specified parameters.
- DELETE removes a subscription.



Attributes

id [read-only, number]

The unique id of a subscription

Elements

metProductType [AIRM / ASPOC / AROME / CAT / CONVECTION_INDEX_GRID / ICING_GRID / ICING_OBJECT / LIGHTNING / METAR / SIGMET / RDT / TAF]

The met product type a client is interested on.

supplyMode [ON_EVENT / SCHEDULE]

The mode that MET-GATE uses to supply subscriptions. *ON_EVENT* mode sends MET Information to the users as soon as new information is available. *SCHEDULE* sends MET information at regular intervals specified by the user in the frequency element.

state [optional, ACTIVE / ENDED]

The state of a subscription.

startingDate [optional, datetime]

The planned time for the start of a subscription.

endingDate [datetime]

The planned time for the end of a subscription.

frequency [optional, number]

The interval time **in minutes** at which a subscription is provided to its user.

publicationMode [AUTOMATED_POLLING / EMAIL_NOTIFICATION]

The mode that MET-GATE uses to supply MET Information to client.

AUTOMATED_POLLING works in automate environment. The MET-GATE sends messages to a broker for notifying a client program that new information is available.

EMAIL_NOTIFICATION sends an email to the client specifying an URL where the MET Information is available.

binding [list(serviceParam)]

The OGC query parameter list.

serviceParam [pair(name, value)]

The OGC key-value pair

name [string]

The name of the OGC query parameter.

value [string]

The value of the OGC query parameter.



Query Parameters

GET and HEAD operations without any query parameters return all subscriptions belonging to the API.

sortField [optional, any of the subscription elements] (to be implemented)
Sort all subscriptions by the specified element.

sortOrder [optional, ASC / DESC] (to be implemented)
Sort all subscriptions in ascending or descending order. By default all subscriptions are sorted in ascending order.

Examples

GET (XML)

Request

```
GET http://claudius.meteo.fr/wcube-dataservice-rest-  
ejb/api/1.0/atmusers/meteofrance/subscriptions HTTP/1.1  
Accept-Encoding: gzip,deflate  
Host: claudius.meteo.fr  
Connection: Keep-Alive  
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)  
Authorization: Basic bWV0ZW9mcmFuY2U6JG1ldGVvLWZyYW5jZTIwMTU=
```

Response

```
HTTP/1.1 200 OK  
Date: Tue, 21 Jul 2015 14:45:12 GMT  
Server: MFWS  
Content-Type: application/xml  
Content-Length: 841  
Keep-Alive: timeout=15, max=299  
Connection: Keep-Alive  
  
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>  
<subscriptions>  
  <subscription id="1">  
    <endDate>2015-08-15T14:25:29Z</endDate>  
    <frequency>1</frequency>  
    <metProductType>SIGMET</metProductType>  
    <nextEventDate>2015-07-21T14:45:13Z</nextEventDate>  
    <binding>  
      <service-param>  
        <name>service</name>  
        <value>WFS</value>  
      </service-param>  
      <service-param>  
        <name>version</name>  
        <value>1.0.0</value>  
      </service-param>  
      <service-param>  
        <name>request</name>  
        <value>GetFeature</value>
```



```
</service-param>
<service-param>
  <name>typeName</name>
  <value>sigmet</value>
</service-param>
<service-param>
  <name>outputFormat</name>
  <value>GML2-MF</value>
</service-param>
</binding>
<publicationMode>AUTOMATED_POLLING</publicationMode>
<startingDate>2015-06-03T14:25:29Z</startingDate>
<state>ACTIVE</state>
<supplyMode>SCHEDULE</supplyMode>
<valid>true</valid>
</subscription>
</subscriptions>
```

POST (XML)

Request

```
POST http://claudius.meteo.fr/wcube-dataservice-rest-
ejb/api/1.0/atmusers/meteofrance/subscriptions HTTP/1.1
Accept-Encoding: gzip,deflate
Content-Type: application/xml
Content-Length: 918
Host: claudius.meteo.fr
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
Authorization: Basic bWV0ZW9mcmFuY2U6JG1ldGVvLWZyYW5jZTIwMTU=
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<subscriptions>
  <subscription id="2">
    <endingDate>2015-09-15T14:25:29Z</endingDate>
    <frequency>1</frequency>
    <metProductType>SIGMET</metProductType>
    <nextEventDate>2015-07-21T14:45:13Z</nextEventDate>
    <binding>
      <service-param>
        <name>service</name>
        <value>WFS</value>
      </service-param>
      <service-param>
        <name>version</name>
        <value>1.0.0</value>
      </service-param>
      <service-param>
        <name>request</name>
        <value>GetFeature</value>
      </service-param>
      <service-param>
        <name>typeName</name>
        <value>sigmet</value>
      </service-param>
      <service-param>
        <name>outputFormat</name>
```



```
<value>GML2-MF</value>
</service-param>
</binding>
<publicationMode>AUTOMATED_POLLING</publicationMode>
<startingDate>2015-06-03T14:25:29Z</startingDate>
<state>ACTIVE</state>
<supplyMode>ON_EVENT</supplyMode>
<valid>true</valid>
</subscription>
</subscriptions>
```

Response

```
HTTP/1.1 201 Created
Date: Tue, 21 Jul 2015 13:31:13 GMT
Server: MFWS
Content-Length: 0
Location: http://claudius.meteo.fr/wcube-dataservice-rest-
ejb/api/1.0/atmusers/meteofrance/subscriptions/2
Keep-Alive: timeout=15, max=299
Connection: Keep-Alive
Content-Type: text/plain
```

PUT (XML)

Request

```
PUT http://claudius.meteo.fr/wcube-dataservice-rest-
ejb/api/1.0/atmusers/meteofrance/subscriptions/1 HTTP/1.1
Accept-Encoding: gzip,deflate
Content-Type: application/xml
Content-Length: 918
Host: claudius.meteo.fr
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
Authorization: Basic bWV0ZW9mcmFuY2U6JG1ldGVvLWZyYW5jZTIwMTU=
```

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<subscriptions>
  <subscription id="1">
    <endingDate>2015-08-15T14:25:29Z</endingDate>
    <frequency>1</frequency>
    <metProductType>SIGMET</metProductType>
    <nextEventDate>2015-07-21T14:45:13Z</nextEventDate>
    <binding>
      <service-param>
        <name>service</name>
        <value>WFS</value>
      </service-param>
      <service-param>
        <name>version</name>
        <value>1.0.0</value>
      </service-param>
      <service-param>
        <name>request</name>
        <value>GetFeature</value>
      </service-param>
    </binding>
  </subscription>
</subscriptions>
```



```
<name>typeName</name>
  <value>sigmet</value>
</service-param>
<service-param>
  <name>outputFormat</name>
  <value>GML2-MF</value>
</service-param>
</binding>
<publicationMode>AUTOMATED_POLLING</publicationMode>
<startingDate>2015-06-03T14:25:29Z</startingDate>
<state>ACTIVE</state>
<supplyMode>SCHEDULE</supplyMode>
<valid>true</valid>
</subscription>
</subscriptions>
```

Response

HTTP/1.1 200 OK
Date: Tue, 21 Jul 2015 14:46:22 GMT
Server: MFWS
Content-Type: application/xml
Content-Length: 767
Keep-Alive: timeout=15, max=299
Connection: Keep-Alive

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<subscriptions>
  <subscription id="1">
    <endingDate>2015-10-15T14:25:29Z</endingDate>
    <frequency>1</frequency>
    <metProductType>SIGMET</metProductType>
    <nextEventDate>2015-07-21T14:45:13Z</nextEventDate>
    <binding>
      <service-param>
        <name>service</name>
        <value>WFS</value>
      </service-param>
      <service-param>
        <name>version</name>
        <value>1.0.0</value>
      </service-param>
      <service-param>
        <name>request</name>
        <value>GetFeature</value>
      </service-param>
      <service-param>
        <name>typeName</name>
        <value>sigmet</value>
      </service-param>
      <service-param>
        <name>outputFormat</name>
        <value>GML2-MF</value>
      </service-param>
    </binding>
    <publicationMode>AUTOMATED_POLLING</publicationMode>
    <startingDate>2015-06-03T14:25:29Z</startingDate>
    <state>ACTIVE</state>
```



```
<supplyMode>ON_EVENT</supplyMode>
<valid>true</valid>
</subscription>
</subscriptions>
```

DELETE (XML)

Request

```
DELETE http://claudius.meteo.fr/wcube-dataservice-rest-
ejb/api/1.0/atmusers/meteofrance/subscriptions/1 HTTP/1.1
Accept-Encoding: gzip,deflate
Host: claudius.meteo.fr
Connection: Keep-Alive
User-Agent: Apache-HttpClient/4.1.1 (java 1.5)
Authorization: Basic bWV0ZW9mcmFuY2U6JG1ldGVvLWZyYW5jZTIwMTU=
```

Response

```
HTTP/1.1 204 No Content
Date: Tue, 21 Jul 2015 14:48:36 GMT
Server: MFWS
Content-Length: 0
Keep-Alive: timeout=15, max=299
Connection: Keep-Alive
Content-Type: text/plain
```

4.2. Automated polling configuration

The MET-GATE, as described in section 4.1, permits to ATM clients to configure a subscription to be treated in an automatic environment. Therefore, when specifying such subscription configuration, the MET-GATE will send the notifications to a notification queue created automatically when registering an ATM user. This procedure will allow ATM programs to connect to their specific queues and so receive and deal with the notifications sent by the MET-GATE in an automatic way.

The MET-GATE provides automatic notification by using any of the following protocols:

- AMQP (port 5672)
- STOMP (port 61613)

Both protocols are available at the end point **http://claudius.meteo.fr**.

The queue name is composed by the static string **“/queue/metgate-broker-notification-”** plus the atm user identifier.

For example, for an ATM client whose registration name is Thales, its associated queue will be **“/queue/metgate-broker-notification-thales”**.

Notification Message

The notification message content is very simple. It is described using JSON format which has only the element URL whose value is the URL where the MET Information is available.



Format Notification Message Content

- {“url”:value, “metproducttype”: “AROME”}

Example

- {"url":"http://localhost:8180//wcube-dataservice-service-webapp/files/73f547fd-158c-4901-886c-1e5a2617dcd8.nc4", "metproducttype":"AROME"}



5. Swim Mater Class Services

This section aims to be a guide to specify how to map the general services WCS and WFS into specific ones by predefining some of the parameters in the OGC requests.

5.1. Airport MET forecast service (CHECK)

In order to obtain Airport MET forecast information, we have to fix the parameter **typeName** and check that property **obsAndFctTime** is greater than **lastResvison** in the WFS request.

- **typeName** = **cdpa**
- **obsAndFctTime** > **lastResvison**

Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=cdpa&count=50&outputFormat=GML3.2-MF&filter=<Filter>
    <PropertyIsGreaterThan>
        <ValueReference>obsorfcsttime</ValueReference>
        <ValueReference>lastrevision</ValueReference>
    </PropertyIsGreaterThan>
</Filter>
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF">
  <Query typeName="cdpa">
    <fes:Filter>
      <fes:PropertyIsGreaterThan>
        <fes:ValueReference>obsorfcsttime</fes:ValueReference>
        <fes:ValueReference>lastrevision</fes:ValueReference>
      </fes:PropertyIsGreaterThan>
    </fes:Filter>
  </Query>
</GetFeature>
```

5.2. Airport MET observation service (CHECK)

This service needs to fix **typeName** and check that property **obsAndFctTime** is greater than **lastResvison** in the WFS request.

- **typeName** = **cdpa**
- **obsAndFctTime** = **lastResvison**



Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=cdpa&count=50&outputFormat=GML3.2-MF&filter=<Filter>
    <PropertyIsEqualTo>
        <ValueReference>obsorfcsttime</ValueReference>
        <ValueReference>lastrevision</ValueReference>
    </PropertyIsEqualTo>
</Filter>
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF">
  <Query typeName="cdpa">
    <fes:Filter>
      <fes:PropertyIsEqualTo>
        <fes:ValueReference>obsorfcsttime</fes:ValueReference>
        <fes:ValueReference>lastrevision</fes:ValueReference>
      </fes:PropertyIsEqualTo>
    </fes:Filter>
  </Query>
</GetFeature>
```

5.3. MET gridded forecast service

This service use the generic WCS. See Section 3.0

5.4. MET Hazard En Route forecast service

This service needs to fix **typeName**, **entitytype** and checks that property **obsAndFctTime** is greater than **lastResvison** in the WFS request.

- typeName = airm
- entitytype = [icing, convection, turbulence]
- obsAndFctTime > lastResvison

Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=airm&count=50&outputFormat=GML3.2-MF&filter=<Filter>
    <And>
```




```
<PropertyIsEqualTo>
  <ValueReference>entitytype</ValueReference>
  <Literal>Convection</Literal>
</PropertyIsEqualTo>
<PropertyIsGreaterThan>
  <ValueReference>obsorfcsttime</ValueReference>
  <ValueReference>lastrevision</ValueReference>
</PropertyIsGreaterThan>
</And>
</Filter>
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF">
  <Query typeName="airm">
    <fes:Filter>
      <fes:And>
        <fes:PropertyIsEqualTo>
          <fes:ValueReference>entitytype</fes:ValueReference>
          <fes:Literal>Convection</fes:Literal>
        </fes:PropertyIsEqualTo>
        <fes:PropertyIsGreaterThan>
          <fes:ValueReference>obsorfcsttime</fes:ValueReference>
          <fes:ValueReference>lastrevision</fes:ValueReference>
        </fes:PropertyIsGreaterThan>
      </fes:And>
    </fes:Filter>
  </Query>
</GetFeature>
```

5.5. MET Hazard En Route observation service

This service needs to fix **typeName**, **entitytype** and checks that property **obsAndFctTime** is equal to **lastResvision** in the WFS request.

- typeName = airm
- entitytype = [icing, convection, turbulence]
- obsAndFctTime = lastResvision

Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=airm&count=50&outputFormat=GML3.2-MF&filter=<Filter>
  <And>
    <PropertyIsEqualTo>
      <ValueReference>entitytype</ValueReference>
      <Literal>Convection</Literal>
```



```
</PropertyIsEqualTo>
<PropertyIsEqualTo>
  <ValueReference>obsorfcsttime</ValueReference>
  <ValueReference>lastrevision</ValueReference>
</PropertyIsEqualTo>
</And>
</Filter>
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF">
  <Query typeName="airm">
    <fes:Filter>
      <fes:And>
        <fes:PropertyIsEqualTo>
          <fes:ValueReference>entitytype</fes:ValueReference>
          <fes:Literal>Convection</fes:Literal>
        </fes:PropertyIsEqualTo>
        <fes:PropertyIsEqualTo>
          <fes:ValueReference>obsorfcsttime</fes:ValueReference>
          <fes:ValueReference>lastrevision</fes:ValueReference>
        </fes:PropertyIsEqualTo>
      </fes:And>
    </fes:Filter>
  </Query>
</GetFeature>
```

5.6. METAR service

This service needs to fix **typeName** in the WFS request.

- typeNanme = metariwxxm

Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=metariwxxm&count=50&outputFormat=GML3.2-MF
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF"
  count="50">
```



```
<Query typeName="metariwxxm">
</Query>
</GetFeature>
```

5.7. TAF service

This service needs to fix **typeName** in the WFS request.

- typeName = tafiwxxm

Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=tafiwxxm&count=50&outputFormat=GML3.2-MF
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF"
  count="50">
  <Query typeName="tafiwxxm">
  </Query>
</GetFeature>
```

5.8. SIGMET service

This service needs to fix **typeName** in the WFS request

- typeName = sigmet

Examples

GET Request

```
http://claudius.meteo.fr/geoserver/metgate/wfs?service=WFS&version=2.0.0&request=GetFeature&typeName=sigmet&count=50&outputFormat=GML3.2-MF
```

POST Request

```
<GetFeature service="WFS" version="2.0.0"
  xmlns="http://www.opengis.net/wfs/2.0"
  xmlns:metgate="http://metgate"
  xmlns:fes="http://www.opengis.net/fes/2.0"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://www.opengis.net/wfs/2.0
http://schemas.opengis.net/wfs/2.0.0/wfs.xsd"
  outputFormat="GML3.2-MF"
```



METEO FRANCE
Toujours un temps d'avance

```
count="50">  
  <Query typeNames="sigmet">  
  </Query>  
</GetFeature>
```



References

- [1] <http://www.opengeospatial.org/standards/wfs>
- [2] <http://geoserver.org/>
- [3] <http://docs.geoserver.org/latest/en/user/services/wfs/reference.html>
- [4] http://external.opengeospatial.org/twiki_public/CoveragesDWG/WcsMetoc
- [5] <http://www.opengeospatial.org/standards/wcs>