



# Project ENDORSE

Energy Downstream Service  
Providing energy components for GMES  
Grant Agreement no 262892

## TUTORIAL ON MACHINE-TO-MACHINE ACCESS TO HELIOCLIM-3 WPS

|  |   |
|--|---|
| <i>Deliverable Number:</i>                 | Internal document   |
| <i>Version Number and Revisions</i>        | v1  |
| <i>WP Number:</i>                          | 3   |
| <i>Authors and Affiliation:</i>            | Benoît GSCHWIND, Lucien WALD, ARMINES   |
| <i>Dissemination Level:</i>                | CO (Confidential, only for members of the consortium, including the Commission Services). |
| <i>Estimated Indicative Person-Months:</i> | 0.5   |
| <i>Activity Type:</i>                      | RTD   |
| <i>Planned Delivery Date:</i>              | November 2011   |
| <i>Actual Delivery Date:</i>               | November 2011   |

## **SUMMARY**

This internal reports documents the access to the current MACC HelioClim-3 product by a machine. The HelioClim-3 product is a NetCDF file containing a time-series of irradiation. Request for a product can be performed automatically by a computer (machine-to-machine access). This report details the steps to do so.

## TABLE OF CONTENTS

|   |   |
|---|---|
| Introduction.....   | 4 |
| Access to HelioClim-3 WPS.....                              | 4 |
| Human-readable description of the HelioClim-3 service:..... | 4 |
| SOAP binding:.....  | 4 |
| WPS binding.....  | 5 |
| GetCapabilities.....  | 5 |
| GetDescribeProcess.....                                     | 5 |
| Request Irradiation Data.....                               | 5 |

## Introduction

The current MACC service offers access to two databases: HelioClim-3 from ARMINES, and SOLEMI from DLR.

Access to the MACC HelioClim-3 products can be done by a standard Web browser. It is not very convenient when many requests are to be made at once or on a regular basis. ARMINES has established means allowing a computer to access HelioClim-3 products without human intervention.

A Web Processing Service (WPS) has been created. Once properly addressed, it returns a NetCDF file containing a time-series of irradiation. This report details the steps to do so.

The MACC HelioClim-3 service provides irradiation data for Europe and Africa. The spatial resolution of this data base is 3 x 3 km<sup>2</sup> at longitude 0° at Equator and decreases towards high latitudes and longitudes. The spatial resolution for Europe ranges from 4 km to 10 km. The temporal resolution is 15 min.

## Access to HelioClim-3 WPS.

### ***Human-readable description of the HelioClim-3 service:***

The HelioClim-3 service has 6 inputs:

- latitude, in decimal form (e.g., 43.78), positive North (ISO standard),
- longitude, in decimal form (e.g., -7.54), positive East (ISO standard),
- first date of period, in the form YYYY-MM-DD (ISO standard),
- last date of period, in the form YYYY-MM-DD (ISO standard),
- elevation above sea level (called altitude here), in integer and in m. If set to -999, the elevation is automatically read for this location in a digital terrain model (DTM). The first choice is the SRTM model. If there is no value for this location, then, a request is made to the GTOPO30, and if there is no value, to the TerrainBase model modified for its use by HelioClim-3,
- time reference: "tu" or "tst" string. 'tu' means UTC time, 'tst' means True Solar Time, i.e., the time system where the time is 12:00 when the sun is at its highest.

The output of service is a netcdf file which contains a time series of 15-min global irradiation on horizontal plane:

- irradiation at the top of atmosphere,
- irradiation that would be observed at ground if the sky were clear, called clear-sky irradiation,
- HelioClim3 irradiation,
- rely, a value relative to the quality of the assessment.

### ***SOAP binding:***

IF you are using SOAP, a WSDL describes the SOAP access at:

[http://toolbox.webservice-energy.org/TOOLBOX/manager?output=xml&cmd=getSrvWSDL&serviceName=Solar\\_Radiation](http://toolbox.webservice-energy.org/TOOLBOX/manager?output=xml&cmd=getSrvWSDL&serviceName=Solar_Radiation)

## WPS binding

To WPS users the service can be reach at the following url:

[http://toolbox.webservice-energy.org:80/TOOLBOX/http/Solar\\_Radiation](http://toolbox.webservice-energy.org:80/TOOLBOX/http/Solar_Radiation)

### GetCapabilities

The service currently supports the POST request only. The GetCapabilities request looks like:

```
<wps:GetCapabilities service="WPS" language="en-US"
xmlns:wps="http://www.opengis.net/wps/1.0.0" />
```

Put this line in a plain text file, named post\_get\_capabilities.xml. Then the request can be made by the following command:

```
curl -o capabilities.xml -H "Content-Type: test/xml" -v -d @post_get_capabilities.xml
http://toolbox.webservice-energy.org:80/TOOLBOX/http/Solar_Radiation
```

The capabilities.xml contains the full description of WPS, as does WSDL for SOAP binding.

### GetDescribeProcess

In the same way we can get the operation description of compute\_irradiation\_helioclim3. To do so put in a plain text file called post\_get\_process\_description.xml the following lines:

```
<wps:DescribeProcess service="WPS" version="1.0.0" language="en"
xmlns:wps="http://www.opengis.net/wps/1.0.0" xmlns:ows="http://www.opengis.net/ows/1.1">
  <ows:Identifier>GetHelioclim3Irradiation</ows:Identifier>
</wps:DescribeProcess>
```

You can send the request to the server with a command line similar to the previous, that is:

```
curl -o helioclim3_describe_process.xml -H "Content-Type: test/xml" -v -d
@WebServices/post_describe_process.xml http://toolbox.webservice-
energy.org:80/TOOLBOX/http/Solar_Radiation
```

The file helioclim3\_describe\_process.xml contains the description of inputs and outputs of the WPS.

### Request Irradiation Data

Finally we can request data to the service in a similar way with the following request:

```
<?xml version="1.0" encoding="UTF-8" standalone="yes"?>
<wps:Execute service="WPS" version="1.0.0" xmlns:ows="http://www.opengis.net/ows/1.1"
xmlns:xlink="http://www.w3.org/1999/xlink" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" xsi:schemaLocation="http://www.opengis.net/wps/1.0.0
../schemas/wps/1.0.0/wpsExecute_request.xsd" xmlns:wps="http://www.opengis.net/wps/1.0.0">
  <ows:Identifier>GetHelioclim3Irradiation</ows:Identifier>
  <wps>DataInputs>
    <wps:Input>
      <ows:Identifier>latitude</ows:Identifier>
      <wps>Data>
        <wps:LiteralData>42.104347826087</wps:LiteralData>
      </wps>Data>
    </wps:Input>
    <wps:Input>
      <ows:Identifier>longitude</ows:Identifier>
      <wps>Data>
```

```

    <wps:LiteralData>0.51594202898551</wps:LiteralData>
  </wps>Data>
</wps:Input>
<wps:Input>
  <ows:Identifier>date_begin</ows:Identifier>
  <wps>Data>
    <wps:LiteralData>2005-02-01</wps:LiteralData>
  </wps>Data>
</wps:Input>
<wps:Input>
  <ows:Identifier>date_end</ows:Identifier>
  <wps>Data>
    <wps:LiteralData>2005-03-04</wps:LiteralData>
  </wps>Data>
</wps:Input>
<wps:Input>
  <ows:Identifier>time_reference</ows:Identifier>
  <wps>Data>
    <wps:LiteralData>tst</wps:LiteralData>
  </wps>Data>
</wps:Input>
<wps:Input>
  <ows:Identifier>altitude</ows:Identifier>
  <wps>Data>
    <wps:LiteralData>-999</wps:LiteralData>
  </wps>Data>
</wps:Input>
</wps>DataInputs>
<wps:ResponseForm>
  <wps:ResponseDocument storeExecuteResponse="true">
    <wps:Output asReference="true" mimeType="application/x-netcdf">
      <ows:Identifier>irradiation</ows:Identifier>
    </wps:Output>
  </wps:ResponseDocument>
</wps:ResponseForm>
</wps:Execute>

```

Put this request in a file call request\_wps\_compute\_irradiation\_helioclim3.xml and use the following command to send the request:

```

curl -o request_status.xml -H "Content-Type: test/xml" -v -d
@WebServices/request_wps_compute_irradiation_helioclim3.xml http://toolbox.webservice-
energy.org:80/TOOLBOX/http/Solar_Radiation

```

The result of this request is not the result but contains the link to the status of the result. The content looks like the following:

```

<?xml version="1.0" encoding="UTF-8"?>
<wps:ExecuteResponse xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:fo="http://www.w3.org/1999/XSL/Format" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" service="WPS"
serviceInstance="http://localhost:8080/TOOLBOX/services/WPSexample/GetCapabilities.xml"
statusLocation="http://toolbox.webservice-energy.org:80/TOOLBOX/manager?
cmd=getServiceResource&serviceName=Solar_Radiation&relativePath=Log/30112011150
936098/status/executeResponse.xml" version="1.0.0" xml:lang="en">
  <wps:Process wps:processVersion="0.1">
    <ows:Identifier>GetHelioclim3Irradiation</ows:Identifier>
    <ows:Title>Compute Irradiation from Helioclim3</ows:Title>
    <ows:Abstract>Compute Solar Irradiation of year 2005 with Helioclim3 Databases</ows:Abstract>
  </wps:Process>

```

```
<wps:Profile/></wps:Process><wps:Status creationTime="2011-11-30T15:09:36Z">
  <wps:ProcessStarted/>
</wps:Status>
</wps:ExecuteResponse>
```

In this response the statusLocation attribute contains the link to the request status:

[http://toolbox.webservice-energy.org:80/TOOLBOX/manager?cmd=getServiceResource&serviceName=Solar\\_Radiation&relativePath=Log/30112011150936098/status/executeResponseStatus.xml](http://toolbox.webservice-energy.org:80/TOOLBOX/manager?cmd=getServiceResource&serviceName=Solar_Radiation&relativePath=Log/30112011150936098/status/executeResponseStatus.xml)

This url is XML encoded. To use it you have to replace “&” by a simple “&”. By this way you access the following content :

```
<?xml version="1.0" encoding="UTF-8"?>
<wps:ExecuteResponse xmlns:wps="http://www.opengis.net/wps/1.0.0"
xmlns:ows="http://www.opengis.net/ows/1.1" xmlns:xlink="http://www.w3.org/1999/xlink"
xmlns:fo="http://www.w3.org/1999/XSL/Format" xmlns:xsi="http://www.w3.org/2001/XMLSchema-
instance" service="WPS"
serviceInstance="http://localhost:8080/TOOLBOX/services/WPSexample/GetCapabilities.xml"
statusLocation="http://toolbox.webservice-energy.org:80/TOOLBOX/manager?
cmd=getServiceResource&serviceName=Solar_Radiation&relativePath=Log/30112011150
936098/status/executeResponseStatus.xml" version="1.0.0" xml:lang="en">
  <wps:Process wps:processVersion="0.1">
    <ows:Identifier>GetHelioclim3Irradiation</ows:Identifier>
    <ows:Title>Compute Irradiation from Helioclim3</ows:Title>
    <ows:Abstract>Compute Solar Irradiation of year 2005 with Helioclim3 Databases</ows:Abstract>
  </wps:Process>
  <wps:Status creationTime="30112011150937718">
    <wps:ProcessSucceeded/>
  </wps:Status>
  <wps:ProcessOutputs>
    <wps:Output>
      <ows:Identifier>irradiation</ows:Identifier>
      <ows:Title>Solar Radiation</ows:Title>
      <ows:Abstract>Monthly average solar irradiation for year 2005</ows:Abstract>
      <wps:Reference href="http://toolbox.webservice-energy.org:80/TOOLBOX/manager?
cmd=getServiceResource&serviceName=Solar_Radiation&relativePath=Log/30112011150
936098/output/irradiation" mime-type="application/x-netcdf"/>
    </wps:Output>
  </wps:ProcessOutputs>
</wps:ExecuteResponse>
```

If the process is finished without error you get this response, and you can find in href attribute the link to the result:

[http://toolbox.webservice-energy.org:80/TOOLBOX/manager?cmd=getServiceResource&serviceName=Solar\\_Radiation&relativePath=Log/30112011150936098/output/irradiation](http://toolbox.webservice-energy.org:80/TOOLBOX/manager?cmd=getServiceResource&serviceName=Solar_Radiation&relativePath=Log/30112011150936098/output/irradiation)

As mentioned previously, the link is XML encoded, so replace “&” to “&” to be able to get the content of the netcdf file.

If you are performing this final step manually, the WebBrowser should display a “save as” dialog, save in the name you want then you can use this netcdf data. If you want another location you have to do another request with updated inputs. This process is not very human friendly, but services are designed to be consumed by machine.