Map4RDF Installation Guide

Preliminaries: dependencies

You should make sure to have the following installed on your computer before starting the installation of Map4RDF:

- Java JDK 1.7 or later
    - http://docs.oracle.com/javase/7/docs/webnotes/install/index.html
    - http://www.oracle.com/technetwork/java/javase/downloads/jdk7-downloads-1880260.html
- GitHub (https://help.github.com/articles/set-up-git)
    - To generate a SSH keys and add in into your github account: (https://help.github.com/articles/generating-ssh-keys)
- Maven (http://maven.apache.org/download.cgi)
- Tomcat 6.0 or later
    - http://tomcat.apache.org/tomcat-6.0-doc/setup.html
    - http://tomcat.apache.org/download-60.cgi

Downloading Map4RDF

Using git in a command line mode

The Map4RDF source code can be obtained with the following command:

```
git clone git@github.com:Fransiro/linked-data-visualization-tools.git
```

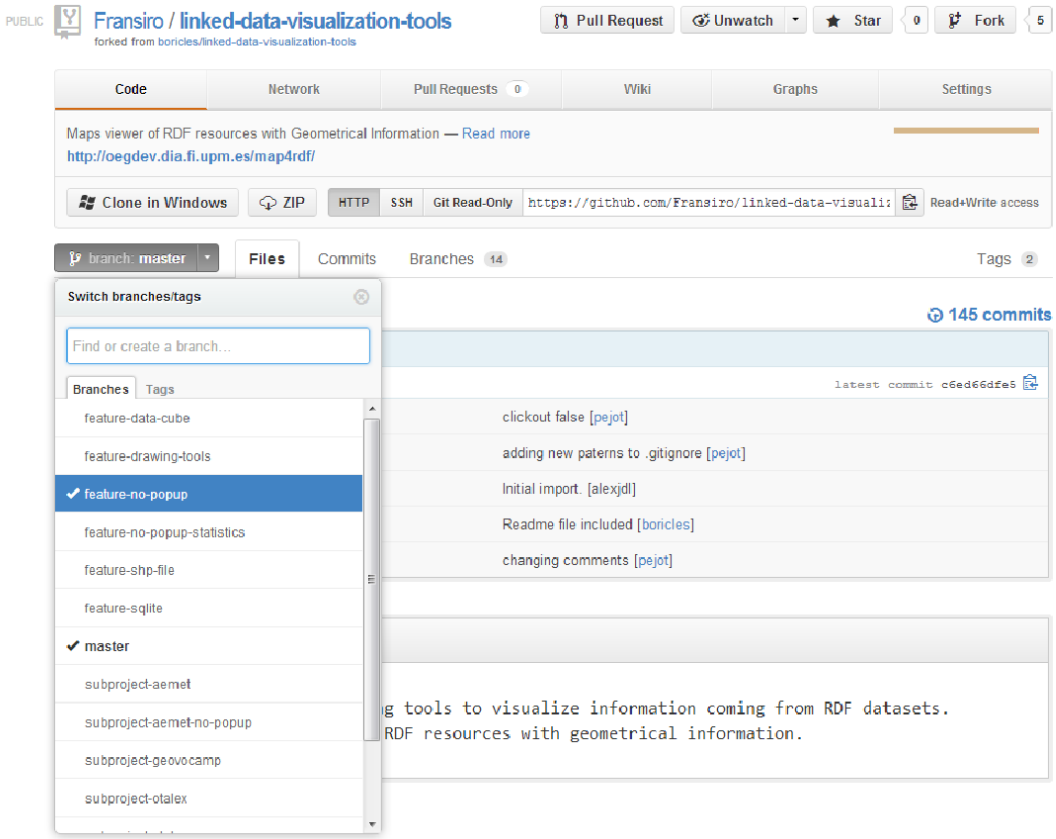
To download a branch:

```
git clone -b <branch name> git@github.com:Fransiro/linked-data-visualization-tools.git
```

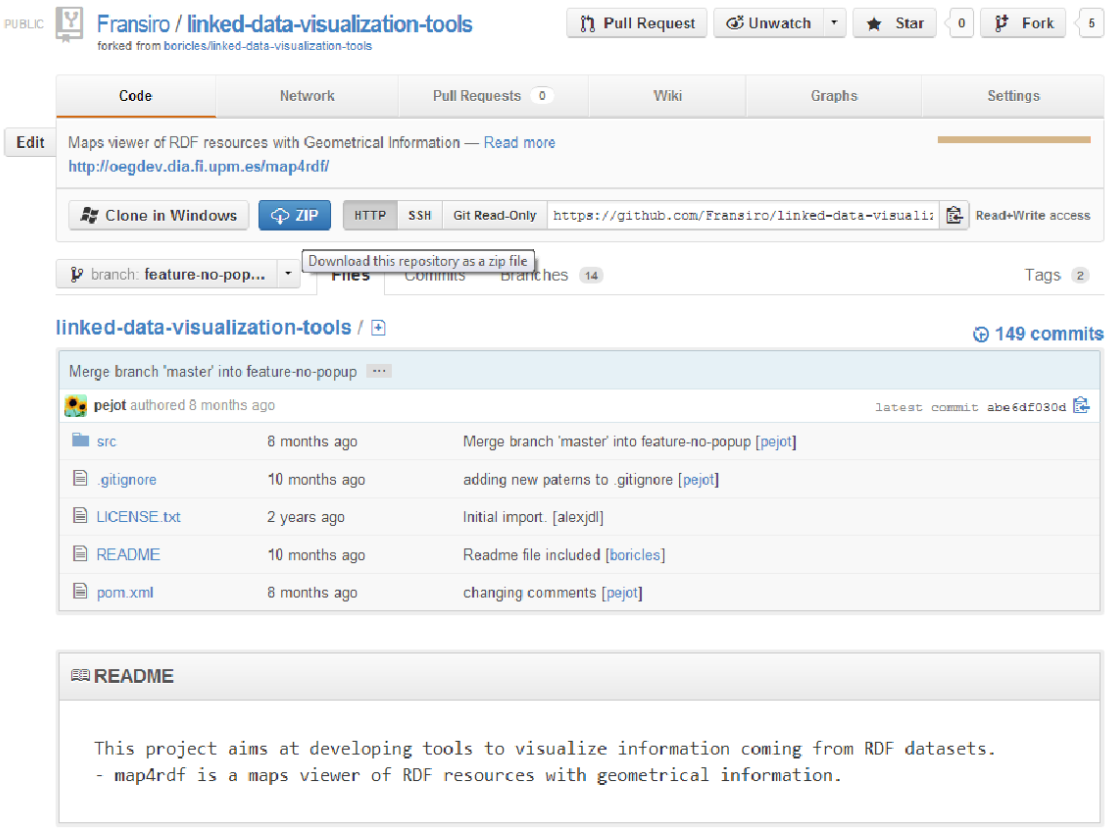
For example: git clone -b update git@github.com:Fransiro/linked-data-visualization-tools.git

Using the GitHub Website

Go to: https://github.com/Fransiro/linked-data-visualization-tools
Select a branch:



Download it (click in a zip button):



Map4RDF installation

Compile Map4RDF

Open a command line window in the Map4RDF download folder.
Execute “mvn install”
If there are no errors during compilation, you can continue.

Deploy Map4RDF in a Tomcat server

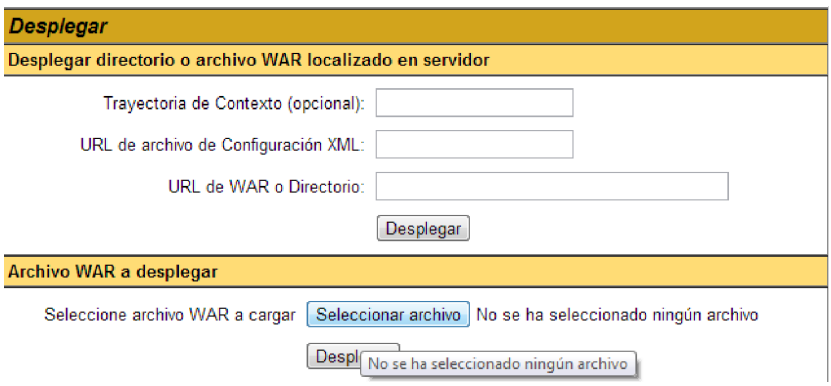
There are two options to deploy Map4RDF in a Tomcat server. You can either copy the folder that is generated during the installation into the Tomcat server webapps folder, or deploy using a war file (recommended).

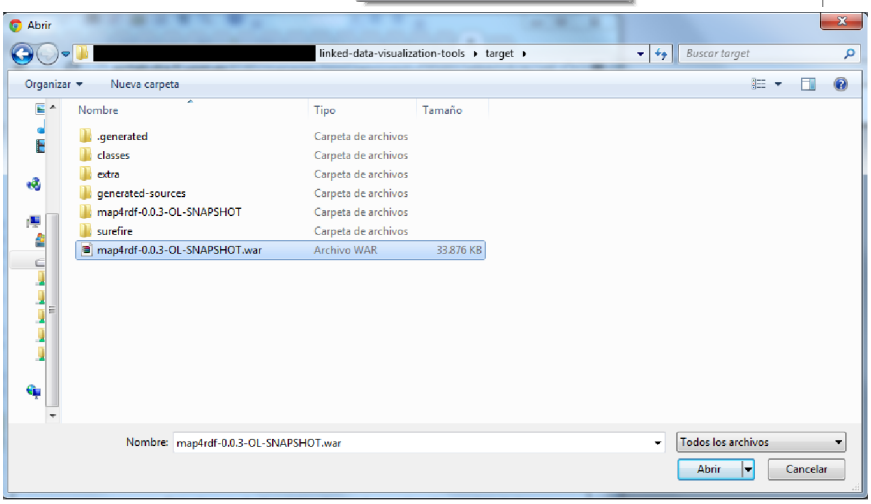
If you want to deploy directly by copying the Map4RDF installation folder, use the following sequence of operations:

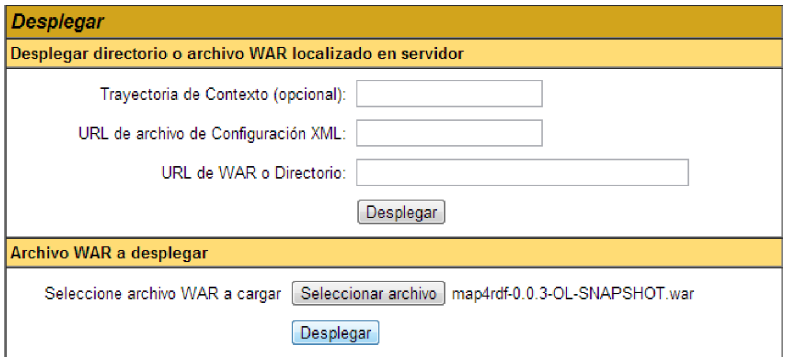
1. In the Map4RDF installation folder enter into the “target” folder.
2. Copy the folder map4RDF-<version>-OL-SNAPSHOT into a “webapps” folder in tomcat server.

However, we recommend you to deploy Map4RDF through the Tomcat manager using a WAR file:

1. Go to: <name-server>:<port>/manager
2. Click on select file, and select the map4RDF-<version>-OL-SNAPSHOT.war file that is available in the folder <Map4RDF folder>/target
3. Click on the deploy button:







Map4RDF Configuration

There are plenty of properties that can be configured in order to have a Map4RDF installation working. Normally, it should suffice with the first set of core properties that come by default in the configuration file available in the installation.

configuration.properties

The **configuration.properties** file for Map4RDF is located at the following folder:
<tomcat-folder>/webapps/<map4RDF folder>/WEB-INF/

The file include the following parameters:

endpoint.url	The SPARQL endpoint where the data to be visualized is available
geometry.model	One of the following: OEG, GEOSPARQL, DBPEDIA, VCARD We expect that these models will be changing in time according to representation model that will be more widespread in the future.
ui.google_maps_api_key	The API Key V3 provided by the Google Maps API, which can be obtained as described in: https://developers.google.com/maps/documentation/javascript/tutor#api_key
facets.automatic	This can be either true or false: • false: the facets to be shown are loaded from the *facets.ttl* file. • true: the facets to be shown are loaded using SPARQL on t endpoint at endpoint.url. See the *facets.ttl* file for more information on how to configure the facets to be shown.
edit_depth	When the edit mode for resources is enabled, it indicates the depth the resource edit mode
rdf_store_path	This is the path where the RDF generated through the editions on t edit mode are stored

spherical_mercator	It indicates the type of map to be shown. It can be either true or fals ● true: sphericals maps (e.g.: google maps, openstreet maps) ● false: WMS maps (e.g.: IDEE maps, openlayers maps, CartoCiudad maps)
default_projection	It indicates the default projection for those points that do not have a explicit projection (e.g., EPSG:4326) More information on projections at http://spatialreference.org/ref/epsg/
map_default_center	Two coordinates separated with "," (first Longitude and then Latitu (e.g., map_default_center=-3.703637,40.416645)
map_zoom_level	Level of map zoom (e.g., map_zoom_level=6)
route_service_timeout_ms	Timeout in milliseconds for the routes service (when using the CartoCiudad service) to search for a route. If time expires, Map4RDF uses the Google route service. The CartoCiudad route service can be disabled by putting 0.
wikipedia_parse_url	URL of the service that parses wikipedia links, so that Map4RDF w concatenate to that URL a wikipedia page that describes a point in the map (wikipedia page need to be a property of point in endpoint) If you do not want to use wikipedia, you can remove it from summary_widgets.
summary_widgets	Ordered (clockwise) list of widgets to present as a summary of each point in the map. For example, summary_widgets =info;wikipedia;buffer;twitter;close;edit;routes;rdf;statistics;type:2 The following widgets are currently available: ● info → Information of a point(Label,latitude, longitude, CR and additional info. See additional_info parameter for more information. ● wikipedia → Wikipedia info (see wikipedia_parse_url) ● buffer → To put a point in buffer mode (Buffer locate point around this with a given distance) ● twitter → To generate a twitter status of this point (see twitter_status_url) ● close → To close summary of point. ● edit → Enter to rdf edit mode of point. ● routes → To add this point in routes mode (Routes mode trace a route from point to a point (And waypoints). ● rdf → To view rdf page of point. ● statistics → To view statistics of point. ● type: NUMBER → available numbers: ○ 0 → Don't move. ○ 1 → Circular movement (automatic radius calculate) ○ 2 → Move of the center to the outside (automatic radius calculate)
statistics_service_url	This is the base URL of the statistics service, in case that we want t

	use it (e.g., statistics_service_url=http://linkeddata2.dia.fi.upm.es:8080/stat/ser s/)
twitter_status_url	Base URL to generate a twitter status (e.g., twitter_status_url=http://twitter.com/?status=)
additional_info	Link to the additional properties files to add in to the info widget. If you want not to use additional info put in configuration.propertie this parameter: "additional_info= " (Space after parameter). Example: additional_info=bikes.properties;other.properties

Additional Info Files

Additional info files are used to provide very specific types of visualisations for some specific types of information that we may want to present in a Map4RDF installation.

The files that are specified in the additional_info parameter must exist in the folder WEB-INF/additional_info. For example if you specify "additional_info=bikes.properties", then the file WEB-INF/additional_info/bikes.properties must exist.

The following parameters must be provided in such a file:

endpoint.url	This parameter specific the endpoint url and parameters to inclu the query in URL and do the query. Example: http://streams.linkeddata.es/citybikes/sparqlstream?query=
query	This parameter specific the query for the endpoint. You can use reserved words for application. The application replace this words for a specific param of marked resource. The reserved words are: !RESOURCE! This is the URI of marked resource. Please remove all '\b' '\r' characters ("New line" characters). Example: PREFIX ssn: <http://purl.oclc.org/NET/ssnx/ssn#> PREFIX qudt: <http://data.nasa.gov/qudt/owl/qudt#> PREFIX bicy: <http://transporte.linkeddata.es/ontology/> PREFIX time: <http://www.w3.org/2006/time#> PREFIX dcterms: <http://purl.org/dc/terms/> PREFIX xsd: <http://www.w3.org/2001/XMLSchema#> SELECT ?avbikes ?freeslots ?tstamp ?obs ?av FROM NAMED STREAM <http://transporte.linkeddata.es/ontology/CityBikes.srdf> [NOW - 300 S] WHERE { ?obs a bicy:FreeBikesObservation. ?obs bicy:inStation <!RESOURCE!>. ?obs ssn:observationResult ?output. ?output bicy:hasAvailableBikesValue ?av. ?av qudt:numericValue ?avbikes. ?output bicy:hasFreeSlotsValue ?fs. ?fs qudt:numericValue ?freeslots. ?obs ssn:observationResultTime ?i . ?i time:inXSDDateTime ?tstamp. }

input_parameters	This parameter specific the reserved works use in query parameter. If you for example use !RESOURCE! you need to specific in hi parameter that use this word. Example: input_parameters=!RESOURCE!
parameters_and_labels	This parameter specific the output parameter of sparql endpoint (Response) and labels for each output parameter to use in info c resource. For specific a output parameter use the next template: OUTPUT_PARAMETER_1:LABELS_1# OUTPUT_PARAMETER_2:LABELS_2 Labels use the next template: LABEL_1@LANGUAGE_1;LABEL_2@LANGUAGE_2 Each parameter need to be separated with "#" character. Each label need to be separated with ";" character. Example: parameters_and_labels=avbikes:Available Bikes@en;Bicicletas disponibles@es#freeslots:Huecos libres@es;Free slots@en#tstamp:Hora de la medición@es;Time stamp@en

The next parameters are for functionality of images limit.

has_images_limit	This parameter dont need exist. This parameter specifies whether this additional info file use the functionality of images limit. Valid values: true, false. Example: has_images_limit=true
images_limit_parameter	This parameter need exist if has_images_limit=true. This parameter specific the variable to analyze a use the functionality of images limit. Example: images_limit_parameter=avbikes
images_limit	This parameter need exist if has_images_limit=true. This parameter specific the inferior and superior limit for put a difference images. This parameter need to have 2 doubles separated with ";". Example: images_limit=3;6

images	This parameter need exist if has_images_limit=true. This parameter specific the different images to us this functionality. This parameter need to have 3 names of images files separated with ";". This images files need to be in es.upm.fi.dia.oeg.map4rdf.map4rdf folder (this folder are in app folder). The images need to have 35px X 35px size. Example: images=bikes_red.png;bikes_yellow.png;bikes_g een.png

The functionality of image limit are:

```
if(images_limit_parameter < images_limit[0]){
        use(images[0]);
}
if(images_limit_parameter > images_limit[0]){
        use(images[2]);
}
if(images_limit_parameter >= images_limit[0] &&
        images_limit_parameter <= images_limit[1]){
        use(images[1]);
}
```

All parameters in this file need to be specific each one in one line.

facets.ttl

The facets.ttl file needs to be non-empty. This file specifies which are the predicates to be used to generate the facets that will appear in the left-hand side of Map4RDF.

An example of basic facets ttl file that uses "rdf:type" as the predicate to use to generate facets (that is, it will use the concepts that have instances in the specified SPARQL endpoint):

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix map4rdf:
        <http://code.google.com/p/map4rdf/wiki/ontology#> .

_:rdfTypeFacet
        a map4rdf:FacetGroup;
        map4rdf:facetPredicate rdf:type;
        rdfs:label "Type"@en;
        rdfs:label "Tipo"@es;
```

An example of a more complex configuration of facets:

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix map4rdf: <http://code.google.com/p/map4rdf/wiki/ontology#> .

_:rdfTypeFacetAire
a map4rdf:FacetGroup;
map4rdf:facetPredicate rdf:type;
rdfs:label "Air"@en;
rdfs:label "Aire"@es;
map4rdf:facet <http://geo.linkeddata.es/ontology/Aeropuerto>;
map4rdf:facet <http://geo.linkeddata.es/ontology/Aer%C3%B3dromo>;
map4rdf:facet <http://geo.linkeddata.es/ontology/Aeroclub>;
.

_:rdfTypeFacetTierra
a map4rdf:FacetGroup;
map4rdf:facetPredicate rdf:type;
rdfs:label "Sand"@en;
rdfs:label "Tierra"@es;
map4rdf:facet
<http://phenomenontology.linkeddata.es/ontology/Catedral>;
map4rdf:facet
<http://phenomenontology.linkeddata.es/ontology/Ruinas>;
map4rdf:facet
<http://phenomenontology.linkeddata.es/ontology/Cueva>;
.

_:rdfTypeFacetAgua
a map4rdf:FacetGroup;
map4rdf:facetPredicate rdf:type;
rdfs:label "Aqua"@en;
rdfs:label "Agua"@es;
map4rdf:facet <http://geo.linkeddata.es/ontology/R%C3%ADo>;
map4rdf:facet <http://geo.linkeddata.es/ontology/Arroyo>;
map4rdf:facet <http://geo.linkeddata.es/ontology/Embalse>;
.

<http://geo.linkeddata.es/ontology/R%C3%ADo>
        rdfs:label "Río"@es;
        rdfs:label "River"@en;
.
<http://geo.linkeddata.es/ontology/Arroyo>
        rdfs:label "Arroyo"@es;
        rdfs:label "Stream"@en;
.
<http://geo.linkeddata.es/ontology/Embalse>
        rdfs:label "Reservoir"@en;
        rdfs:label "Embalse"@es;
.

<http://phenomenontology.linkeddata.es/ontology/Catedral>
        rdfs:label "Catedral"@en;
.
<http://phenomenontology.linkeddata.es/ontology/Cueva>
        rdfs:label "Coves"@en;
        rdfs:label "Cueva"@es;
.
```

```
<http://phenomenontology.linkeddata.es/ontology/Ruinas>
        rdfs:label "Ruinas"@es;
        rdfs:label "Ruins"@en;
.

<http://geo.linkeddata.es/ontology/Aeropuerto>
        rdfs:label "Airport"@en;
        rdfs:label "Aeropuerto"@es;
.
<http://geo.linkeddata.es/ontology/Aer%C3%B3dromo>
        rdfs:label "Aerodrome"@en;
        rdfs:label "Aeródromo"@es;
.
<http://geo.linkeddata.es/ontology/Aeroclub>
        rdfs:label "Aeroclub"@en;
.
```

This file specific a 3 box of facet type and 3 facets in each box.

Facets need to be defined in the file.

If facets.automatic = true , the application search in endpoint the facetPredicate “rdf:type”; for each box. This option is for endpoints that have more than one facetPredicate type and find all.

Example image of this facets ttl file with facets.automatic=false (Browser language is spanish (@es)):

Facets

Agua

- [ ] Arroyo
- [ ] Embalse
- [ ] Río

Aire

- [ ] Aeroclub
- [ ] Aeropuerto
- [ ] Aeródromo

Tierra

- [ ] Catedral
- [ ] Cueva
- [ ] Ruinas

Map Configuration

The file for maps configuration is: /WEB-INF/maps/maps.properties

This file contains two properties:

spherical_maps	This property is for spherical_mercator=true In this property you specify the files to include into application. Each file is a or map. Single Map Configuration will be seen later. The specific files need found in /WEB-INF/ maps/spherical_maps/ Example: spherical_maps=openstreet.properties;google_normal.properties In this example these files exists: WEB-INF/maps/spherical_maps/openstreet.properties WEB-INF/maps/spherical_maps/google_normal.properties
flat_maps	This property is for spherical_mercator=false In this property you specify the files to include into application. Each file is a or map. Single Map Configuration will be seen later. The specific files need found in /WEB-INF/ maps/flat_maps/ Example: flat_maps=idee_todas.properties;cartociudad_todas.properties In this example these files exists: WEB-INF/maps/flat_maps/idee_todas.properties WEB-INF/maps/flat_maps/cartociudad_todas.properties

Single Map Configuration:

url	This property is for WMS and OSM map type, specify the service url to obta images. In WMS type is obligatory and in OSM type is optional(Use default OSM service). Example in OSM: url=http://a.tile.openstreetmap.org/${z}/${x}/${y}.png Example in WMS: url=http://www.idee.es/wms-c/IDEE-Base/IDEE-Base
labels	This property specific the labels for client. Is the name that appear in right-up corner of the map. Labels use the next template: LABEL_1@LANGUAGE_1;LABEL_2@LANGUAGE_2 Each label separated with “;” character. Example: labels=Open Street Mapas@es;Open Street Maps@en

type	This property specify the type of map service. This property have 3 possible values: WMS => WMS map service OSM => OSM map service Google => Google map service Example: type=OSM
layers	This property is for WMS type, specify the layer that can be use. In service U include the param for this specific layer. Each WMS service has it owns laye Example: layers=IGNBaseTodo url=http://www.ign.es/wms-inspire/ign-base
attribution	This property include the text(as HTML) to the right-down corner in map. Example: attribution=Maps provided by ID
set_resolutions	This property is a boolean that specify if application set his resolutions in ma use default value. Example: set_resolutions=true
projection	This property specify EPSG projection that use the map. See the next link for available EPSGs: http://spatialreference.org/ref/epsg/ Example: projection=EPSG:4258
set_max_extends	This property is a boolean that specify if application set his bounds in map or default value. Example: set_max_extends=true
format	This property specify the image output format in WMS type. Only WMS typ Example: format=image/png
transition_effect	This property specify if the application set the transition effect: “RESIZE” or Is a boolean. Example: transition_effect=true
zoom_levels	This property specify the maximum zoom levels for the map. Is a integer. Example: zoom_levels=20
gtype	This property specify the Map Google type (in version 3). Only for Google type. The available types are: G_HYBRID_MAP G_NORMAL_MAP G_SATELLITE_MAP G_TERRAIN_MAP See google documentation for further information. Example: gtype=G_NORMAL_MAP
spherical_mercator	This property is only for OSM and Google type, specify if the map is spheric a boolean. Example: spherical_mercator=true