**MQTTConnector**

This software is a complement to the Helio platform that generates RDF using the data obtained from the mqtt communication protocol which allows users to rely on mqtt assignments to generate the RDF and then use other Helio features to improve or publish the RDF

**Quick Start**

To allow Helio to use this add-on, follow these steps:

* Create an assignment using the Helio specification
* In the assignment, specify any of the MQTT connectors of this add-in
* Obtain the MQTTConnector jar file and its dependencies folder, then place it in the Helio add-ons folder
  + You can compile the project as specified below or download the compiled dependency
  + Run Helio normally

**MQTTConnector arguments**

A list indicating the name of the client, the IP address, the number of arguments to be seen and the name of the topics to which we want to subscribe.

**MQTTConnector example mapping**

{

"datasources" : [

{

"id" : "MQTT Datasource",

"type" : "JsonDatasource",

"arguments" : ["$.[\*]"],

"connector" : {

"arguments" : ["demonio","tcp://10.3.141.1","2","universidad/laboratorio/temperatura"],

"type" : "MQTTConnector"

}

}

],

"resource\_rules" : [

{

"id" : "MQTT Temperature",

"datasource\_ids" : ["MQTT Datasource"],

"subject" : "http://localhost:8080/lab/property/temperature",

"properties" : [

{

"predicate" : "http://iot.linkeddata.es/def/core#hasValue",

"object" : "http://localhost:8080/lab/property/temperature/value/[hashCode(REGEXP\_REPLACE({$.timestamp},'\\s\*','\_'))]",

"is\_literal" : "False"

}

]

]

**Compiling the plugin**

To complile the *jar* of this plugin run the following command:

mvn clean install

A folder called dist will be created containing the compiled *jar* and a folder with dependencies called mqtt; both must be pasted in the folder contaning the plugins of Helio in order to use this plugin.