

## How to create a Hashtag Location Map with CreateLocationMap.jar

### Step 1: Download reference city names and coordinates from Geonames (txt file)

<http://download.geonames.org/export/dump/>

Its important that this file complies to the standard data format from Geonames:

Columns: (1) GeonamesId (2) Official name (3) Official name ascii (4) alternative names (separated by ,)  
(5) latitude (6) longitude (7) ..  
\* \* Example Line 1847947 Shingū Shingu Schingu,Shingu,Shingui,Shingū,Sing,Singu,Singū,Синг  
33.73333 135.98333 P PPL JP 43  
31619 7 Asia/Tokyo 2012-01-19

### Step 2: Export Tweets from TCAT (csv file)

### Step 3: Set Parameters in ParseLocationTweet.properties-File

Change input parameters for the script:

filepathGeonamesFile=C:/java/tcat\_addon/JP.txt (Filepath to Geonames File)  
filepathTweetFile=C:/java/tcat\_addon/data.csv (Filepath to CSV file containing Tweets)  
printHTML=1 (Choose whether you want to print a human readable version to see which locations have been detected [0=no / 1=yes])  
printLocationHashtagMap=1 (Choose whether you want to print a location hashtag map)  
printGPSLocationMap=1 (Choose whether you want to print a map between mentioned location and tweet gps tag)  
findLocationsInHashtag=1 (should the algorithm detect locations within hashtags? E.g. #save\_fukushima ?)  
noOfTweetToProcess=100 (how many tweets should be processed, choose 0 for all)  
columnTweetText=6 (indicate the index of the column in the Tweet-file (csv) which contains the Tweet Text, start counting from left with 0)  
columnTweetId=0 (index of the column which contains TweetId)  
columnUserId=2 (index of the column which contains the UserId)  
columnLat=9 (index of the column which contains the Tweet GPS tag-Latitude)  
columnLng=10 (index of the column which contains the Tweet GPS tag-Longitude)  
columnTweetTime=1 (index of the column which contains the Tweet Time)

### Step4: Run the script

Java -Dfile.encoding=UTF-8 -jar CreateLocationHashtagMap.jar

Note that the properties file has to reside in the same folder as the .jar file!

The script produces max 3 Files depending on the settings, which will be stored in the same location as the Tweetfile:

- [inputfilename]\_GPS.csv (Mentioned Location, Tweet-GPS Tag Matching)
- [inputfilename]\_LHM.csv (Location Hashtag Matching)
- [inputfilename]\_result.html (human readable version)

### Step5: Convert result into a Gephi Map

Go to: <http://tools.medialab.sciences-po.fr/table2net/> to convert the csv into a gephi file

- 1) Choose e.g. the data\_LHM.csv as input File
- 2) Choose the network type: Bi-Partite network

- 3) Which column defines the first type of Nodes?
  - a. Select "hashtag"
  - b. Select "Semicolon-separated"
- 4) Which column defines the second type of Nodes?
  - a. Select "locationId"
  - b. Select "One expression per cell"
- 5) Do you want attributes for the *second type* of nodes?
  - a. Select "LocationLabel", "lat", "lng"
- 6) Build the network file
- 7) Open the file in Gephi
- 8) Go to the Data Laboratory
- 9) Select Duplicate Column "Lat", Change the column type to Float click OK
- 10) Select Duplicate Column "Lng", Change the column type to Float click OK
- 11) Go to "Overview"
- 12) Select the Layout "GeoLayout"(you have to install this plugin before the first use: (Tools → Plugins → available Plugins → GeoLayout → install))
- 13) For the attribute "Latitude" Select the column we just created ("lat (type\_2) copy")
- 14) For the attribute "Longitude" Select the column we just created ("lng (type\_2) copy")
- 15) Select Run
- 16) Go to Partition → press the refresh button (2 green arrows) → select "apply"
- 17) Go to Filters → Attributes → Equal → Type (String) Node → drag this element into the queries field below
- 18) Type in "locationId" for pattern and press "Select", thereby we only select the nodes which represent locations which just have been positioned based on their coordinates
- 19) Right click on the selected nodes and press "Settle" thereby we lock the layout options for the nodes which just have been located
- 20) Now you can choose another Layout to arrange the other nodes, in this case hashtags around the locations, e.g. Force Atlas 2 → Run
- 21) You now have created a hashtag location map from your tweet data csv..  
Where locations mentioned in tweets are positioned based on their real geographic locations, and hashtags which co-occur within the same tweet are linked to these locations.