



www.hotmaps-project.eu

Data upload on GitLab

Mostafa Fallahnejad, Sara Fritz

4 December 2017 – V04



Funded by the Horizon 2020 programme
of the European Union



1 INTRODUCTION AND OVERVIEW.....	3
2 STRUCTURE ON GITLAB.....	4
3 DATA FORMATS	5
3.1 Raster Data	5
3.2 Vector Data.....	6
3.3 Tabular Data.....	7
4 DATAPACKAGE.JSON.....	7
4.1 Explanation of the attributes.....	8
4.2 Data package templates.....	10
4.2.1 Data Package Template for Raster Layers.....	11
4.2.2 Data Package Template for Vector Layers	13
4.2.3 Data Package Template for CSV files	15
5 LICENCES	16
REFERENCES	16

1 Introduction and Overview

This document is a short version of the Data Management Plan with some additional information. The information here should be sufficient to guarantee a smooth data provision on GitLab for using in the toolbox. The key points of this document are summarized here. Please read the following chapters for detailed instructions and examples.

This document can be summarized in following points:

- The repository name should be a unique name, written in lower-case and containing only alphanumeric characters along with ".", "_" or "-" characters
- Each repository contains a "README.md" file, a "datapackage.json" file and a "data" folder.
- README.ms should be in Markdown format and provide basic information such as dataset title, content description, methodology/process for data generation, and references to input data sources.
- Raster data must be provided in GeoTiff format.
- User "ETRS89 / ETRS-LAEA EPSG: 3035" for projecting your data.
- Coordinate of top-left corner of your map should be a multiplicand of the resolution of a map.
- Consider the use of raster layer with integer values along with a [transformation](#) factor if it reduces the size of your raster file significantly.
- For the compression of the raster files use these settings: ZLEVEL=9, COMPRESS=DEFLATE, PREDICTOR=1, TILED=YES, TFW=YES.
- Use git Large File Storage (git-lfs) for uploading raster files.
- Use the international system of units (SI units) for your data. You may use metric prefix for the units (Kilo, Mega, Giga, etc.).
- Vector data must be provided in geojson format.
- Instead of providing a vector data for NUTS and LAU regions, you may use a CSV file with NUTS or LAU codes.
- To be consistent, use the NUTS and LAU layers available in Hotmaps GitLab repository.
- CSV files may contain geographic data in form of NUTS or LAU2 codes.
- Each Hotmaps repository should contain one data package file describing the repository and the data in it.
- The optional attributes within the body of templates are highlighted in **yellow**. You can delete them from your data package if you do not want to provide such information.
- The "[case-sensitive](#)" attributes are highlighted in **turquoise** and must be filled if they are relevant; otherwise, you can leave them blank. They **should not be deleted** from your data package file.
- All other attributes must be filled in and provided in your data package file.

2 Structure on GitLab

Each data repository of Hotmaps should have a unique name, written in lower-case and containing only alphanumeric characters along with ".", "_" or "-" characters. These repositories should have a "README.md" file, a "data" folder and a "datapackage.json" file. Fig. 1 shows schematically the structure of Hotmaps GitLab repositories.

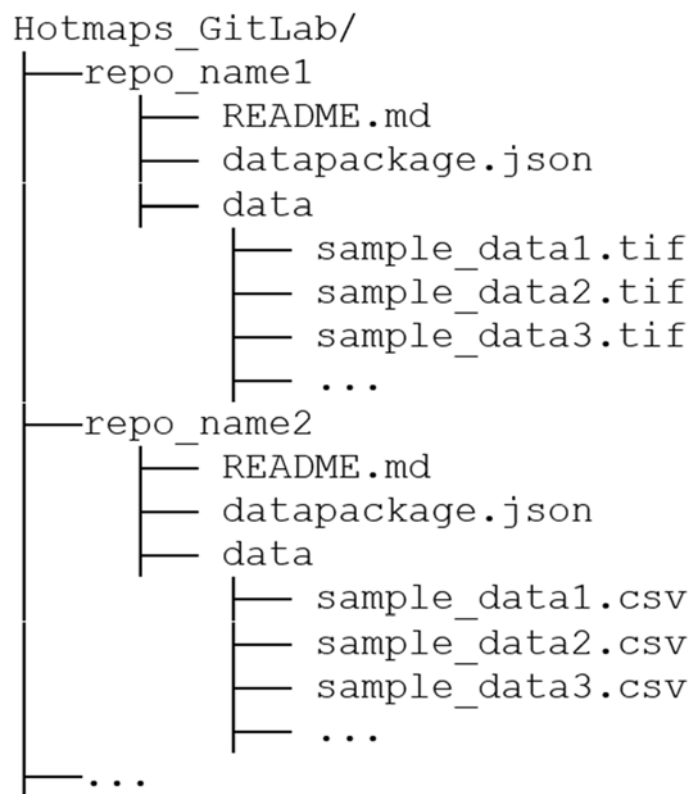


Fig. 1 Structure of Hotmaps GitLab Repository

The "README.md" file should provide basic information such as dataset title, content description, methodology/process for data generation and references to input data sources. The data that is uploaded to a repository should be placed in "data" folder. **The "datapackage.json" is used as metadata that can be read both by human and machine.** The Hotmaps toolbox obtains some important information such as data units or multiplying factor of the raster files from data package file.

Summary of key points

- *The repository name should be a unique name, written in lower-case and containing only alphanumeric characters along with ".", "_" or "-" characters.*
 - *Each repository contains a "README.md" file, a "datapackage.json" file and a "data" folder.*
 - *README.ms should be in Markdown format and provide basic information such as dataset title, content description, methodology/process for data generation, and references to input data sources.*
-

3 Data formats

One of the aims in developing Hotmaps toolbox is to use it for mapping the heating and cooling energy. Therefore, a major part of default data sets will be spatial data. Vector and raster data can be used for these purposes. Tabular data, on the other hand, can be used for both non-spatial data as well as spatial data (using NUTS/LAU codes).

In order to minimize the efforts for reading and understanding the data, and ease the interaction between the Hotmaps toolbox and the data warehouse, a set of data formats are defined. These formats should be used for uploading the default data sets onto the Hotmaps GitLab repository.

3.1 Raster Data

The desired data format for the raster data is **GeoTIFF**. For the raster data, we use the Coordinate Reference System (CRS) of "**ETRS89 / ETRS-LAEA**" with the Authority ID of "**EPSG:3035**". Depending on the resolution of your map, the top-left corner of your raster should have a coordinate that is a multiplicand of the map's resolution, e.g. in case of 1ha resolution it can be $(x, y) = (4212300, 3004200)$ and in case of 1x1km resolution it can be $(x, y) = (4212000, 3004000)$. This leads to a one-to-one relation pixels from different layers in mathematical operations and ease the comparison of layers after the aggregation or disaggregation of them.

The use of integer values along with a multiplying factor can significantly reduce the size of raster layers. Therefore, please consider to use the integer values for such cases. The multiplying factor must be provided in data package file of the repository under "resources

→ raster → transformation”. For more detail on data package files of raster layer refer to section [4.2.1](#). Furthermore, use the international system of units (SI units) for your data. You may use a metric prefix for the units (Kilo, Mega, Giga, etc.).

In order to reduce the size of the raster layers, please use a compressed binary format (GeoTiff generated with the following options: ZLEVEL=9, COMPRESS=DEFLATE, PREDICTOR=1, TILED=YES, TFW=YES) for generating raster layers.

Upload of large GeoTiff rasters on GitLab might be problematic. In such cases, the repository should use a git dedicate extension called: git Large File Storage¹ (git-lfs) that has been specifically developed to efficiently handle large binary files.

Summary of key points

- *Raster data must be provided in GeoTiff format.*
 - *Use “ETRS89 / ETRS-LAEA EPSG: 3035” for projecting your data.*
 - *Coordinate of top-left corner of your map should be a multiplicand of the resolution of a map.*
 - *Consider to use a raster layer with integer values along with a transformation factor if it significantly reduces the size of your raster file.*
 - *For the compression of the raster files use these settings: ZLEVEL=9, COMPRESS=DEFLATE, PREDICTOR=1, TILED=YES, TFW=YES.*
 - *Use git Large File Storage (git-lfs) for uploading raster files.*
 - *Use the international system of units (SI units) for your data. You may use metric prefix for the units (Kilo, Mega, Giga, etc.).*
-

3.2 Vector Data

The desired data format for the vector data is “geojson” since it can be easily implemented in the web browser². The most frequently used administrative boundaries in Hotmaps toolbox are NUTS 0, NUTS 1, NUTS 2, NUTS 3 and LAU 2. Hotmaps uses “NUTS 2013” and “communes 2013” (LAU 2) versions with Coordinate Reference System (CRS) of “ETRS89 / ETRS-LAEA” and

¹ Look at <https://git-lfs.github.com>

² Vector data of NUTS or LAU regions can be also provided in CSV format. The only condition is to provide the NUTS or LAU codes along with other data.

the Authority ID of “EPSG:3035”. Although these layers are publicly available, we strongly ask you to use the ones which are provided on Hotmaps GitLab repository. There are two reasons for this:

1. There are different versions of administrative boundary layers and therefore, different borders and divisions. Furthermore, the IDs or projections may also alter in different versions;
2. Administrative boundary layers are often provided with different scales, e.g. 1:1 Million or 1:20 Million. The use of different scales increases the potential for mismatches in aggregation and disaggregation processes.

Summary of key points

- *Vector data must be provided in geojson format.*
 - *Instead of providing a vector data for NUTS and LAU regions, you may use a CSV file with NUTS or LAU codes.*
 - *To be consistent, use the NUTS and LAU layers available in Hotmaps GitLab repository.*
-

3.3 Tabular Data

Tabular data must be provided in CSV format. This format can also be used to provide data for the NUTS or LAU regions.

4 Datapackage.json

Data package is used to describe a set of data. A data package consists of:

- Metadata that describes the structure and contents of the package,
- Resources such as data files that form the contents of the package.

It can be used for any type of data. The key benefits of using data package files are their simplicity, understandable both for human and machine, and reusability.

In this chapter, a short explanation on frequently used attributes in data package file is provided. The explanations and instructions are extracted from Frictionless recommendations

(see reference section). Subsequently, the data package templates which should be used in Hotmaps repository will be provided. In case of doubts or ambiguity in explanations, please follow the Frictionless specifications.

4.1 Explanation of the attributes

The attributes that are used in a data package file are not limited to the ones that are explained here. One may define additional attributes where it makes sense. Table 1 explains the attributes in a way that are relevant for Hotmaps repositories.

Table 1 Frequently used attributes in data package file

Attribute	Description
<code>name</code>	Every repository should have a name. This MUST be lower-case and contain only alphanumeric characters along with ".", "_", "-" or "-" characters. This name should be unique and may not be used by other repositories. The first element will define the spatial resolution of the data set (e.g. NUT{0 1 2 3} or LAU{1 2}). Then we will use the keywords defined in the wiki-page to describe the dataset. So for example the H&C demand of the building sector at NUTS3 could be: NUTS3_building_HC_residential_demand.
<code>datapackage_version</code>	It shows the version of data package file. This is independent of the version of data sets.
<code>title</code>	A string providing a title or one sentence description for this repository.
<code>description</code>	A plain text explaining the data in repository.
<code>version</code>	A version that showing the version of the data in repository. It should conform to the symantic versioning containing MAJOR.MINOR.PATCH parts, e.g. 0.3.0.
<code>id</code>	A globally unique ID that distinguishes the data in repository with all other data sets. This can be in for of UUID or DOI.
<code>created</code>	Datetime that the package was created.
<code>keywords</code>	Keywords with which one can address the content of the repository. This eases the search for the package in catalogues.

Attribute	Description
<code>licenses</code>	The license under which the package is provided. It should be in array format and contain a name property and/or a path property. It may contain a title property.
<code>sources</code>	The sources for the provided data. Must be entered if available/relevant.
<code>contributors</code>	The people or organizations who contributed to this data package.
<code>resources</code>	Describes the data that is provided in the repository.

The information provided under the “resources” attribute addresses the files inside the “data” folder (raster, vector and csv files). Therefore, all sub-attributes under the category of “resources”, such as name, title, path, etc., also refer to these files. Table 2 describes some of these sub-attributes.

Table 2 Sub-attributes of “resources” for different data formats

Data Type	Sub-attributes of “resources”	Description
All	<code>name</code>	This MUST be lower-case and contain only alphanumeric characters along with ".", "_" or "-" characters. This name should be unique and may not be used by other repositories.
	<code>title</code>	A string providing a title or one sentence description of the file.
	<code>path</code>	Relative path in the repository. E.g.: “data/test.csv”
	<code>description</code>	A plain text explaining the file. This should be markdown formatted.
	<code>temporal</code>	Showing the time attribute of the file. It should be in datetime format (look at here)
	<code>profile</code>	Can be “raster-package” or “vector-package” or “tabular-data-resource”
	<code>format</code>	Can be “tiff” or “geojson” or “csv”
	<code>bytes</code>	Showing size of the file on disk in bytes
	<code>sources</code>	Showing the sources used for this file
	<code>licenses</code>	The license under which a file is provided. For the details that should be included in license, consider the template provided in the next section or use this Link .
Raster	<code>unit</code>	Unit of the map (e.g. MWh)

Data Type	Sub-attributes of “resources”	Description
	<code>epsg</code>	Projection
	<code>transformation</code>	Gives a factor and an operator for applying on raster to get the correct value.
	<code>extent</code>	Shows the coordinate of lower-left and upper-right corners of the map.
	<code>resolution</code>	Resolution of map in X and Y directions.
	<code>dimension</code>	Shows the number of rows and columns of a raster.
	<code>origin</code>	Shows the coordinate of the upper-left corner of the raster
	<code>nodata</code>	How “no data value” is handled.
Vector	<code>statistics</code>	Shows minimum, maximum, mean and standard deviation of the content of the raster.
	<code>geometry_type</code>	e.g. point, linestring, polygon, etc.
	<code>epsg</code>	Projection
	<code>schema</code>	Contains field names, types and units.
	<code>number_of_features</code>	Number of features in map
CSV	<code>extent</code>	Extent of the map
	<code>encoding</code>	e.g. “utf-8”
	<code>schema</code>	Geofields: start and end of geometric columns non-geofields: information regarding non-geometric fields More information in here .
	<code>dialect</code>	Delimiter and use of double quote in entries.

4.2 Data package templates

Here the templates for the data package file are presented. The aim is to keep the explanations minimal and at the same time descriptive. A repository may contain more than one data set, e.g. monthly heating degree days raster files. It should be noted that only one data package file for each repository should be provided and NOT one data package file per one input file.

The data providers are welcomed to use further attributes; however, it is not mandatory. In order to clarify the use of optional attributes, some examples are added to the templates. These examples are highlighted in “yellow”. The yellow lines can be completely deleted from your data package file. Additionally, there are attributes that can be interpreted either as optional or mandatory. However, they must remain in the data package file, though empty! In this document, they are called “case-sensitive” attributes. The case-sensitive attributes can be distinguished by a “turquoise” highlight. A good example of case-sensitive

attributes is the source of the data set. The source **MUST** be provided if the data is taken from another source. On the other hand, it can be skipped and left blank if the data is an outcome of own calculations.

Summary of key points

- *Each Hotmaps repository should contain one data package file describing the repository and the data in it.*
 - *The optional attributes within the body of templates are highlighted in yellow. You can delete them from your data package if you do not want to provide such information.*
 - *The case-sensitive attributes are highlighted in turquoise and must be filled if they are relevant; otherwise, you can leave them empty. They should not be deleted from your data package file.*
 - *All other attributes must be filled in and provided in your data package file.*
-

4.2.1 Data Package Template for Raster Layers

```
{
  "name": "eu_hdm",
  "profile": "raster-data-resource"
  "datapackage_version": [
    "1.0-beta"
  ],
  "title": "Heat density map of EU28 + Switzerland",
  "description": "The map shows the demand density in in each ha
with GJ/km2 unit",
  "version": "0.3.0",
  "id": "87855e0e-e505-3483-ba7a-a3211a6f4982",
  "created": "2017-10-11",
  "keywords": [
    "EU28",
    "EUROPE",
    "heat map",
    "raster"
  ],
  "licenses": [
    {
      "name": "other-pd",
      "path":
"http://licenses.opendefinition.org/licenses/other-pd.json",
      "title": "Other (Public Domain)"
    }
  ]
}
```



```
    },
  ],
  "sources": [
    {
      "name": "Corine Land Cover 2012",
      "path": "http://land.copernicus.eu/pan-european/corine-land-cover/view"
    },
    {
      "name": "Imperviousness 2012",
      "path": "http://land.copernicus.eu/pan-european/high-resolution-layers/imperviousness/imperviousness-2012"
    }
  ],
  "contributors": [
    {
      "title": "Andreas Mueller",
      "email": "mueller@eeg.tuwien.ac.at",
      "role": "author",
      "organization": "tu wien"
    },
    {
      "title": "Mostafa Fallahnejad",
      "email": "fallahnejad@eeg.tuwien.ac.at",
      "role": "author",
      "organization": "tu wien"
    }
  ],
  "resources": [
    {
      "name": "heat_tot_curr_density.tif",
      "title": "Heat density map of EU28 + Switzerland on basis year 2012",
      "path": "data/heat_tot_curr_density.tif",
      "description": "explain the data",
      "temporal": {
        "start": "2012-01-01 00:00:00",
        "end": "2012-12-31 24:00:00"
      },
      "format": "tif",
      "bytes": "111992005",
      "unit": "GJ/km2",
      "raster": {
        "epsg": "3035",
        "num_of_bands": 1,
        "extent": {
          "minx": 944000.0,
          "miny": 938000.0,
          "maxx": 6528000.0,
          "maxy": 5414000.0
        },
        "resolution": {
          "xRes": 100.0,
          "yRes": -100.0
        }
      }
    }
  ]
}
```



```
    },
    "dimension": {
      "columns": 55840,
      "rows": 44760
    },
    "origin": [
      944000.0,
      5414000.0
    ],
    "nodata": 0.0,
    "transformation": {
      "operator": "/",
      "value": 100.0
    },
    "band1": {
      "statistics": {
        "min": 22.000003814697266,
        "max": 162278.09375,
        "mean": 166.73078955654867,
        "Stdev": 240.63384010292674
      }
    }
  }
}
]
```

4.2.2 Data Package Template for Vector Layers

```
{
  "name": "nuts",
  "profile": "vector-data-resource",
  "datapackage_version": [
    "1.0-beta"
  ],
  "title": "nuts regions",
  "description": "This repository contains maps of administrative borders in different NUTS levels",
  "version": "1.0.0",
  "id": "87855e0e-e505-3483-ba7a-a3211a6f4982",
  "created": "2017-10-11",
  "keywords": [
    "admin border",
    "Europe",
    "nuts"
  ],
  "licenses": [
    {
      "name": "unknown",
```



```
        "path": "",
        "title": ""
    },
    "sources": [
        {
            "name": "",
            "path": ""
        }
    ],
    "contributors": [
        {
            "title": "",
            "email": "",
            "role": "",
            "organization": ""
        }
    ],
    "resources": [
        {
            "name": "nuts3.geojson",
            "title": "Heat density map of EU28 + Switzerland on basis
year 2012",
            "path": "data/nuts3.geojson",
            "description": "explain the data",
            "format": "geojson",
            "bytes": "2813",
            "vector": {
                "geometry_type": "Polygon",
                "epsg": "3035",
                "schema": [
                    {
                        "name": "NUTS_ID",
                        "type": "string",
                        "unit": ""
                    },
                    {
                        "name": "STAT_LEVL_",
                        "type": "int",
                        "unit": ""
                    },
                    {
                        "name": "SHAPE_AREA",
                        "type": "float",
                        "unit": ""
                    }
                ]
            },
            "number_of_features": 1480,
            "extent": {
                "minx": -2824432.1198082,
                "miny": -3076239.8068296723,
                "maxx": 10026108.83400277,
                "maxy": 5415981.729751202
            }
        }
    ]
}
```

```

    }
  ]
}

```

4.2.3 Data Package Template for CSV files

```

{
  "name": "load_profile",
  "profile": "tabular-data-resource",
  "datapackage_version": [
    "1.0-beta"
  ],
  "title": "hourly load profile nuts 2 level",
  "description": "The repository contains hourly heat load profile
in NUTS 2 level",
  "version": "0.3.0",
  "id": "87855e0e-e505-3483-ba7a-a3211a6f4982",
  "created": "2017-10-11",
  "keywords": [
    "load profile",
    "EUROPE",
    "nuts 2"
  ],
  "licenses": [
    {
      "name": "",
      "path": "",
      "title": ""
    }
  ],
  "sources": [
    {
      "name": "",
      "path": ""
    }
  ],
  "contributors": [
    {
      "title": "Sara Fritz",
      "email": "fritz@eeg.tuwien.ac.at",
      "role": "author",
      "organization": "tu wien"
    }
  ],
  "resources": [
    {
      "name": "nuts2_lp.csv",
      "title": "hourly load profile nuts 2 level",
      "path": "data/ nuts2_lp.csv",
      "description": "explain the data",
      "temporal": {

```



```
    "start": "2012-01-01 00:00:00",  
    "end": "2012-12-31 24:00:00"  
  },  
  "format": "csv",  
  "bytes": "5836",  
  "encoding": "utf-8",  
  "schema": {  
    "fields": [  
      {  
        "name": "nuts3",  
        "type": "string",  
        "unit": ""  
      },  
      {  
        "name": "hour",  
        "type": "integer",  
        "unit": ""  
      },  
      {  
        "name": "load",  
        "type": "float",  
        "unit": "MJ"  
      }  
    ]  
  },  
  "dialect": {  
    "delimiter": ",",  
    "doubleQuote": true  
  }  
}
```

5 Licences

To be completed

References

<http://frictionlessdata.io/data-packages/>

<http://frictionlessdata.io/specs/data-resource/>

<http://frictionlessdata.io/specs/table-schema/>