

Technical Tuesdays

Leaflet **and** LaTeX

Justin van Dijk // Balamurugan Soundararaj
j.t.vandijk@ucl.ac.uk // s.bala@ucl.ac.uk

Technical Tuesdays

Introduction
15 Oct

R Scripting
29 Oct

JavaScript
19 Nov

Version Control
03 Dec

*nix Shell
22 Oct

Python
12 Nov

Databases
26 Nov

Leaflet and LaTeX
10 Dec



Mapping

R (tmap, ggplot2)

Python (GeoPandas / Pandas / Matplotlib)

Q-GIS

ArcGIS Desktop / Pro

Mapping

R (tmap, ggplot2)

Python (GeoPandas / Pandas / Matplotlib)

Q-GIS

ArcGIS Desktop / Pro

 **Static**

Webmapping

OpenLayers

Leaflet.js

D3

MapBox Studio

ArcGIS Online

Tableau

Webmapping

OpenLayers

Leaflet.js

D3

MapBox Studio

ArcGIS Online

Tableau

Interactive



Webmapping

Data

‘slippy’ maps

Leaflet.js

GeoJSON

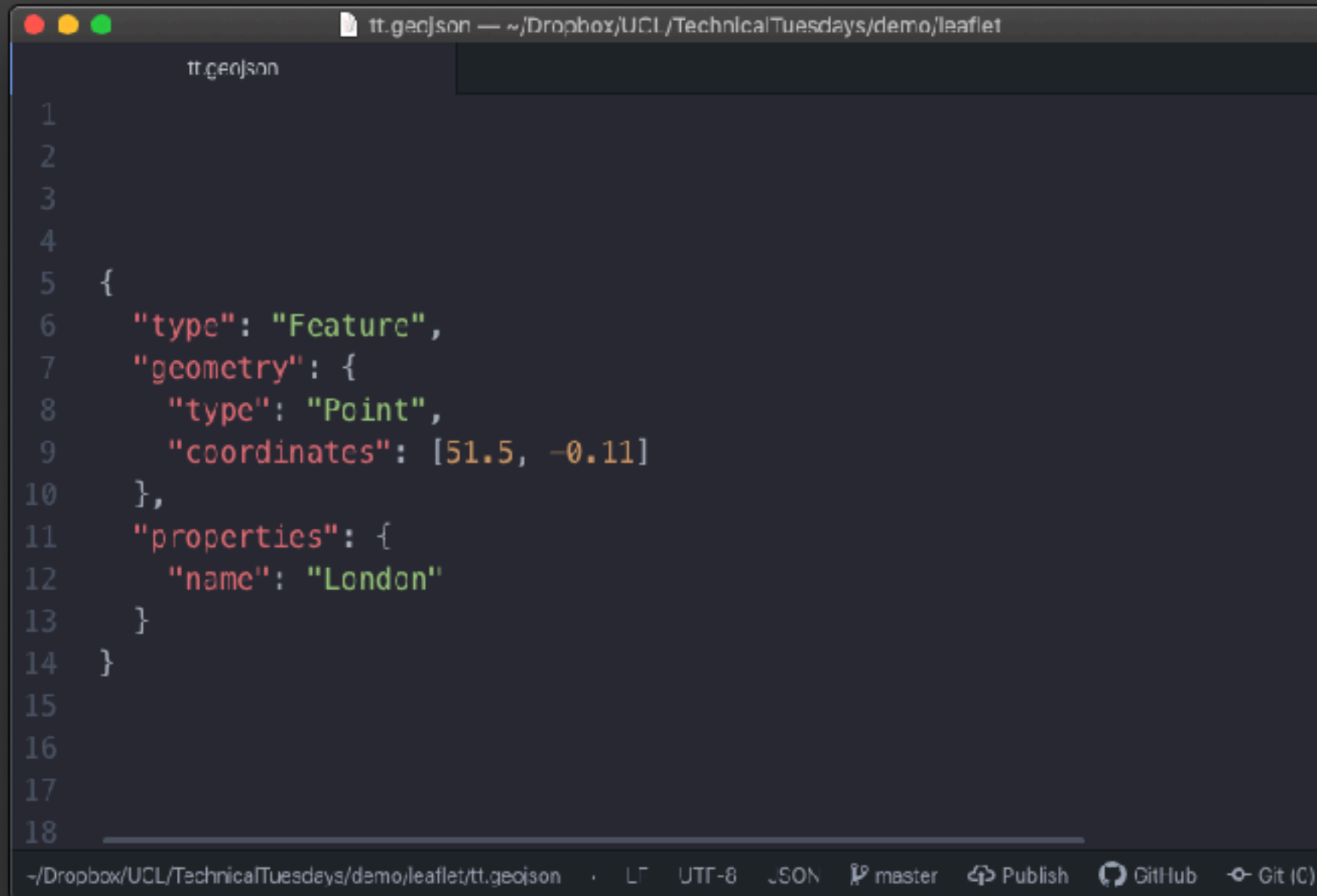
JSON specification

Specify geographical data in an consistent way

Non-geographical attributes

Always uses WGS84 and decimal degrees

GeoJSON

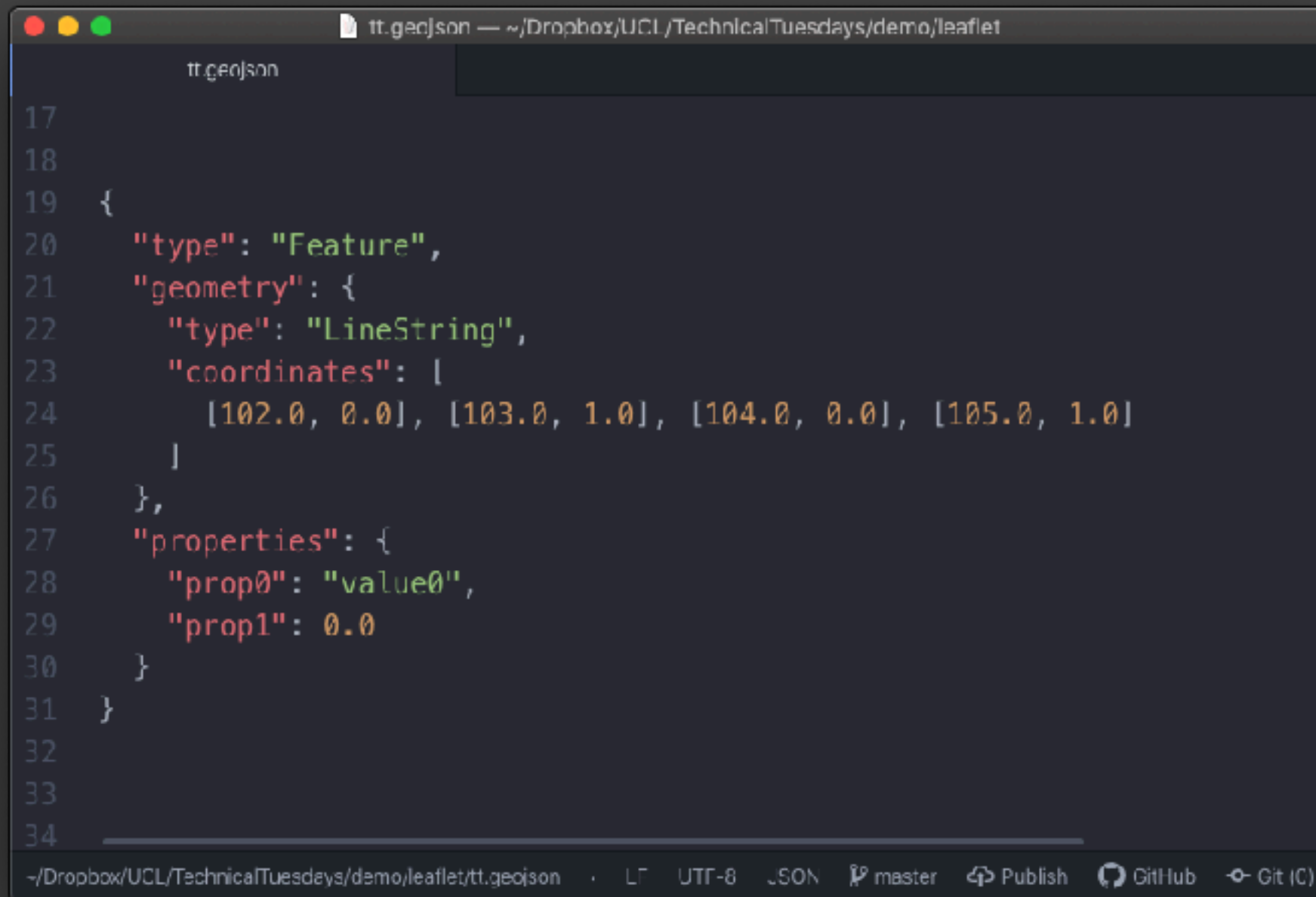


The image shows a code editor window with a dark theme. The title bar at the top reads "tt.geojson — ~/Dropbox/UCL/TechnicalTuesdays/demo/leaflet". The editor has a tab labeled "tt.geojson". On the left side, line numbers 1 through 18 are visible. The code is a GeoJSON Feature object, formatted with syntax highlighting: strings are in red, keywords like "type" are in green, and arrays are in orange. The code is as follows:

```
1
2
3
4
5 {
6   "type": "Feature",
7   "geometry": {
8     "type": "Point",
9     "coordinates": [51.5, -0.11]
10  },
11  "properties": {
12    "name": "London"
13  }
14 }
15
16
17
18
```

At the bottom of the editor, a status bar displays the file path "~/Dropbox/UCL/TechnicalTuesdays/demo/leaflet/tt.geojson", encoding "UTF-8", format "JSON", branch "master", and buttons for "Publish", "GitHub", and "Git (C)".

GeoJSON

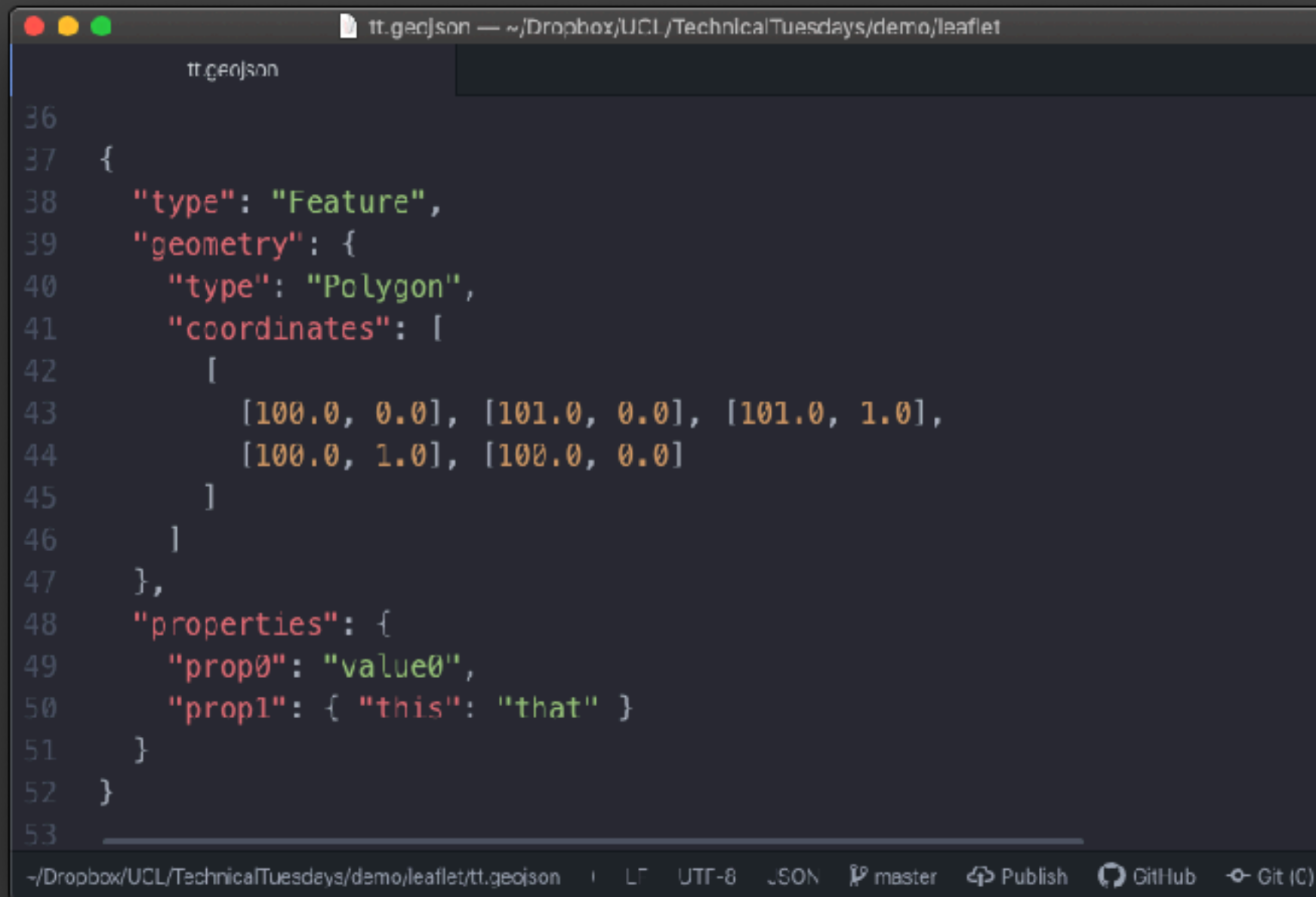


The image shows a code editor window with a dark theme. The title bar at the top reads "tt.geojson — ~/Dropbox/UCL/TechnicalTuesdays/demo/leaflet". The editor has a tab labeled "tt.geojson". The code is a GeoJSON Feature object, with line numbers 17 through 34 visible on the left. The JSON is formatted with syntax highlighting: strings are in red, keys in green, and numbers in orange. The code defines a Feature with a LineString geometry and two properties.

```
17
18
19 {
20   "type": "Feature",
21   "geometry": {
22     "type": "LineString",
23     "coordinates": [
24       [102.0, 0.0], [103.0, 1.0], [104.0, 0.0], [105.0, 1.0]
25     ]
26   },
27   "properties": {
28     "prop0": "value0",
29     "prop1": 0.0
30   }
31 }
32
33
34
```

The status bar at the bottom shows the file path: "~/Dropbox/UCL/TechnicalTuesdays/demo/leaflet/tt.geojson". It also includes icons for file encoding (UTF-8), format (JSON), branch (master), and actions (Publish, GitHub, Git (C)).

GeoJSON



The image shows a code editor window with a dark theme. The title bar at the top reads "tt.geojson — ~/Dropbox/UCL/TechnicalTuesdays/demo/leaflet". The editor contains a JSON object representing a GeoJSON Feature. The code is as follows:

```
36
37 {
38   "type": "Feature",
39   "geometry": {
40     "type": "Polygon",
41     "coordinates": [
42       [
43         [100.0, 0.0], [101.0, 0.0], [101.0, 1.0],
44         [100.0, 1.0], [100.0, 0.0]
45       ]
46     ]
47   },
48   "properties": {
49     "prop0": "value0",
50     "prop1": { "this": "that" }
51   }
52 }
53
```

The status bar at the bottom of the editor displays the file path: `~/Dropbox/UCL/TechnicalTuesdays/demo/leaflet/tt.geojson`. It also includes icons and labels for encoding (UTF-8), format (JSON), branch (master), and actions like Publish, GitHub, and Git (C).

TopoJSON

Extension of GeoJSON

Compact

Relative coordinates

Shared nodes, arcs defined only once — topology encoded

TopoJSON

```
t:geojson -- ~/Dropbox/UCI/TechicalTuesdays/demos/leaflet
53
54 {
55   "type": "Topology",
56   "transform": {
57     "scale": [1, 1],
58     "translate": [0, 0]
59   },
60   "objects": {
61     "two-squares": {
62       "type": "GeometryCollection",
63       "geometries": [
64         {"type": "Polygon", "arcs": [[0, 1]], "properties": {"name": "Left_Polygon"}},
65         {"type": "Polygon", "arcs": [[2, -1]], "properties": {"name": "Right_Polygon"}}
66       ]
67     },
68     "one-line": {
69       "type": "GeometryCollection",
70       "geometries": [
71         {"type": "LineString", "arcs": [3], "properties": {"name": "Under_LineString"}}
72       ]
73     },
74     "two-points": {
75       "type": "GeometryCollection",
76       "geometries": [
77         {"type": "Point", "coordinates": [0, 0], "properties": {"name": "Origine_Point"}},
78         {"type": "Point", "coordinates": [0, -1], "properties": {"name": "Under_Point"}}
79       ]
80     }
81   },
82   "arcs": [
83     [[1, 2], [0, -2]],
84     [[1, 0], [-1, 0], [0, 2], [1, 0]],
85     [[1, 2], [1, 0], [0, -2], [-1, 0]],
86     [[0, -1], [2, 0]]
87   ]
88 }
89
```

~/Dropbox/UCI/TechicalTuesdays/demos/leaflet/t:geojson 43/2

LF UTF-8 JSON master Publish GitHub OR ID

‘slippy’ maps

Images should be served as tiles on a grid

Amalgamation of individual requested raster or vector files

‘slippy’ maps



‘slippy’ maps



‘slippy’ maps

Pre-rendered rasters (e.g. Mapnik with Python bindings)

Vector tiling (tiles, but vectors)

Leaflet

Open-source JavaScript library

Mobile-friendly interactive maps

Light weight (about 38 KB of JS)

Simplicity

Resources

Leaflet.js

<https://leafletjs.com/index.html>

JavaScript

https://www.w3schools.com/js/js_intro.asp

Demo

LaTeX

/ləˈtɛx/ LAH-tekh or /leɪtɛx/ LAY-tekh

Where does LaTeX come from?



Printing Press

Typesetting



Workspace

Wordprocessing

Why use Latex?

Consistency

Production Quality documents

Programmatically generate documents

Sensible cross-referencing

Where can I start using Latex?

Demo

Examples

Overleaf

<https://www.overleaf.com/latex/templates>

My PhD Thesis

https://github.com/sbm kvp/phd_thesis/raw/master/thesis.pdf

Questions

Feedback!