Web Mapping & Analysis

Data Architecture & Formats

Dani Arribas-Bel

Today

- Spatial Data
- Spatial Data Formats
- Selecting the right format

Spatial Data

How we represent the world in a computer

- Vector
- Raster

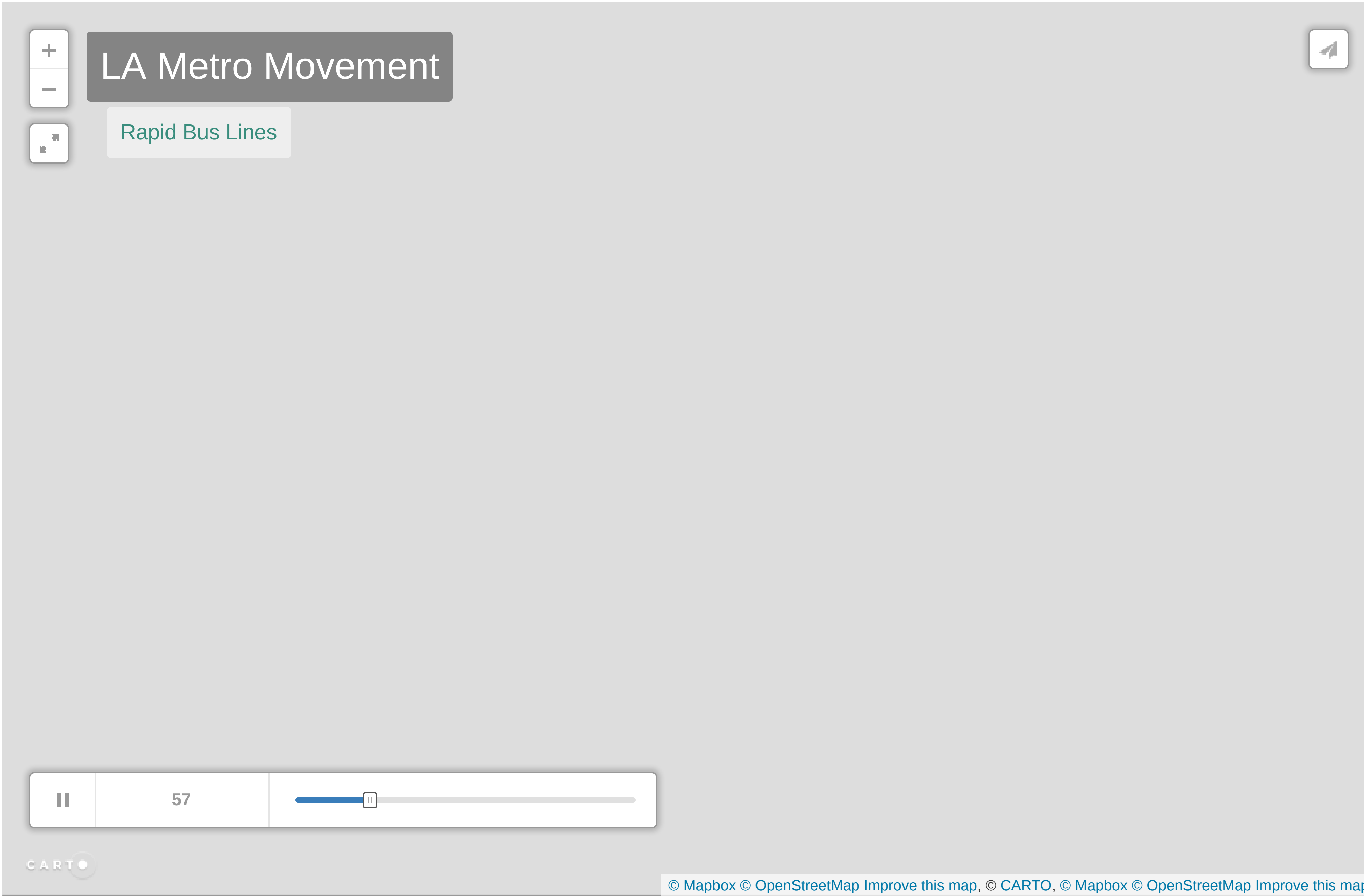
Vector

Represent each entity with a shape or geometry.

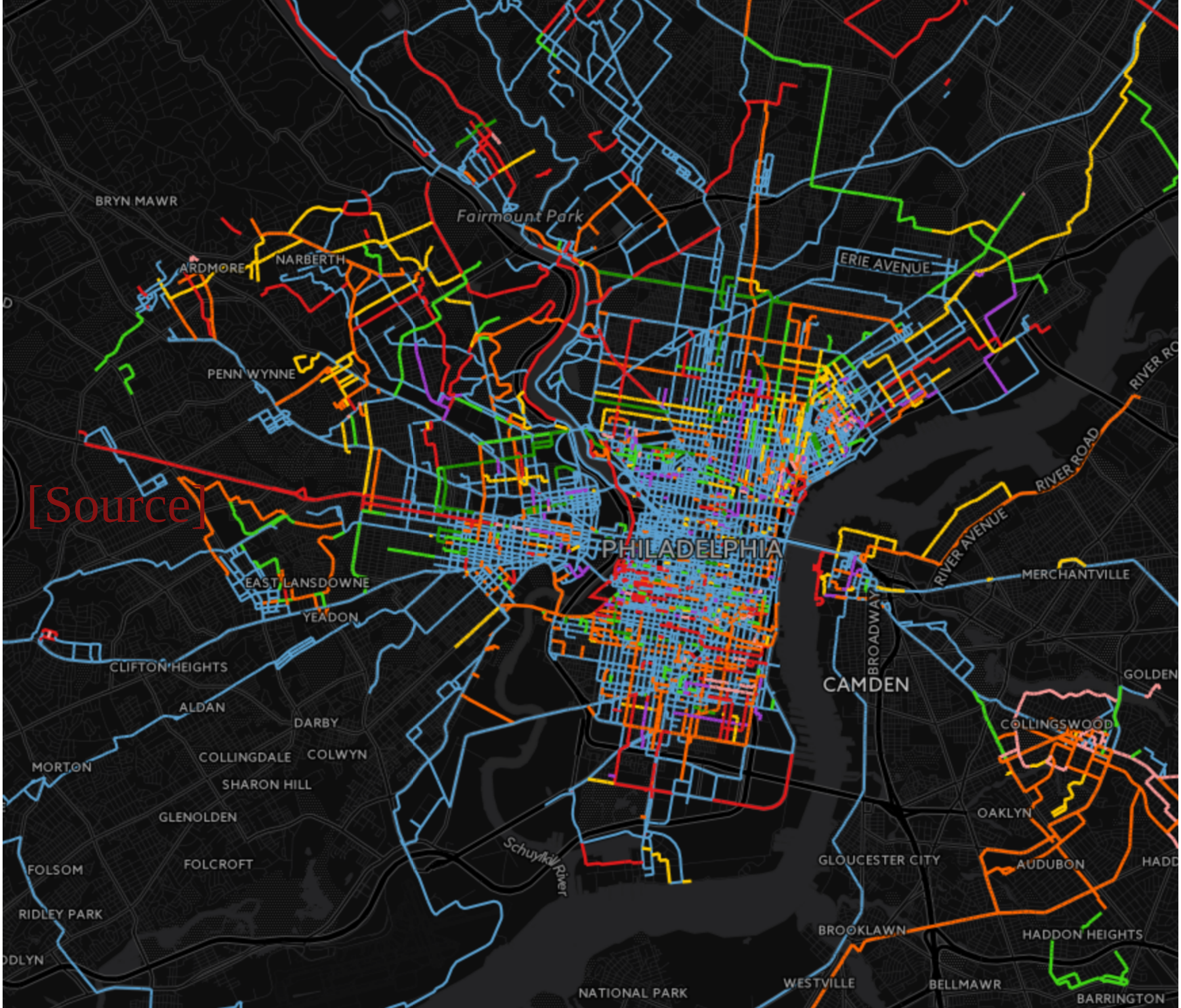
Simple features:

- (Multi-)Points
- (Multi-)Lines
- (Multi-)Polygons

Divide space into a **finite** set of entities



Map created by d9a


BRYN MAWR
Fairmount Park
ARDMORE
NARBERTH
ERIE AVENUE
PENN WYNNE
RIVER RO
RIVER ROAD
RIVER AVENUE
PHILADELPHIA
MERCHANTVILLE
EAST LANSDOWNE
YEADON
BROADWAY
CLIFTON HEIGHTS
CAMDEN
GOLDEN
ALDAN
DARBY
COLLINGSWOOD
COLLINGDALE
COLWYN
MORTON
SHARON HILL
GLENOLDEN
OAKLYN
FOLSOM
FOLCROFT
Schuylkill River
GLOUCESTER CITY
AUDUBON
HADD
RIDLEY PARK
BROOKLAWN
HADDON HEIGHTS
DDLYN
WESTVILLE
BELLMAWR
NATIONAL PARK
BARRINGTON

[Source]

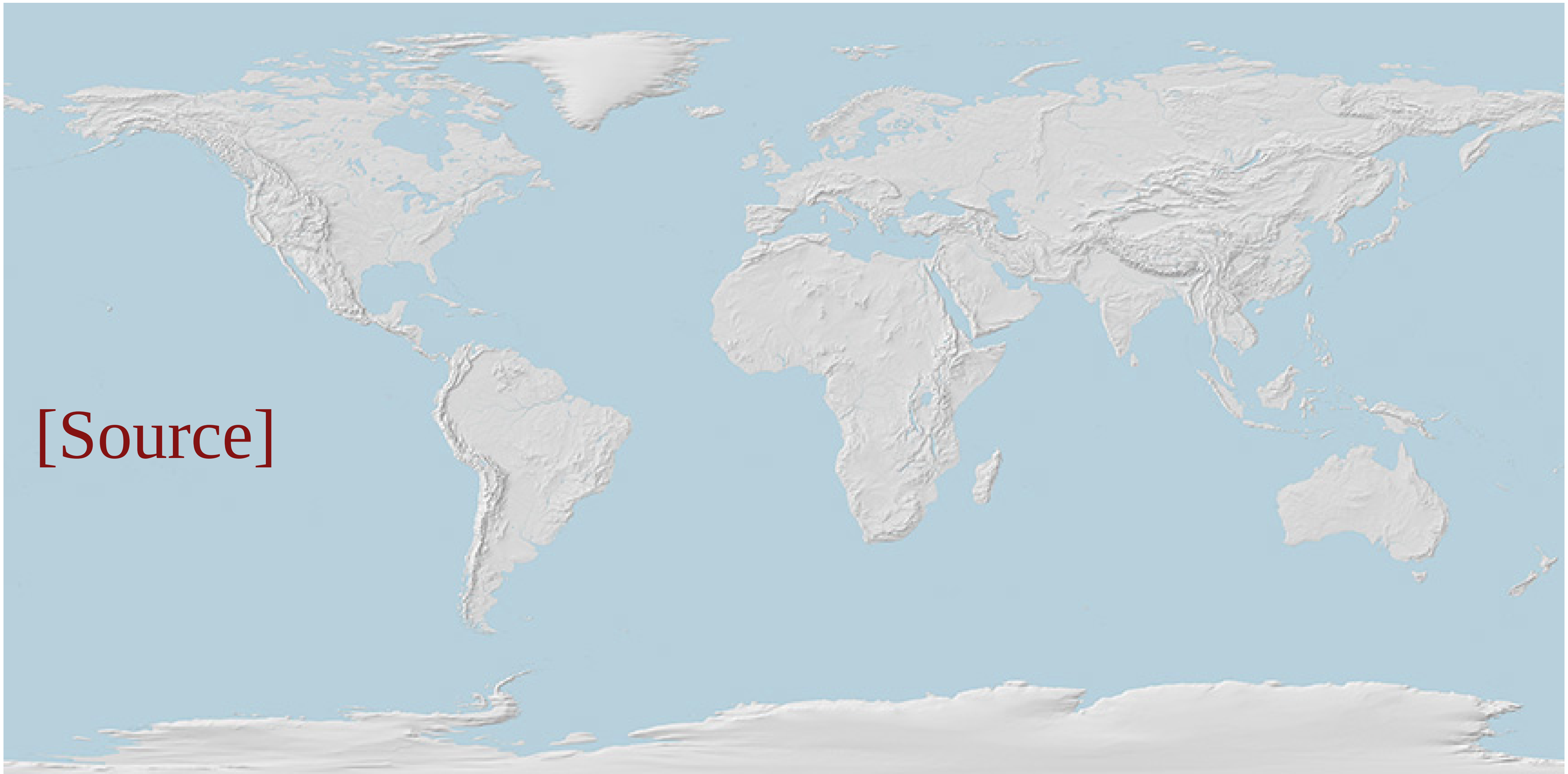

[Source]

Raster

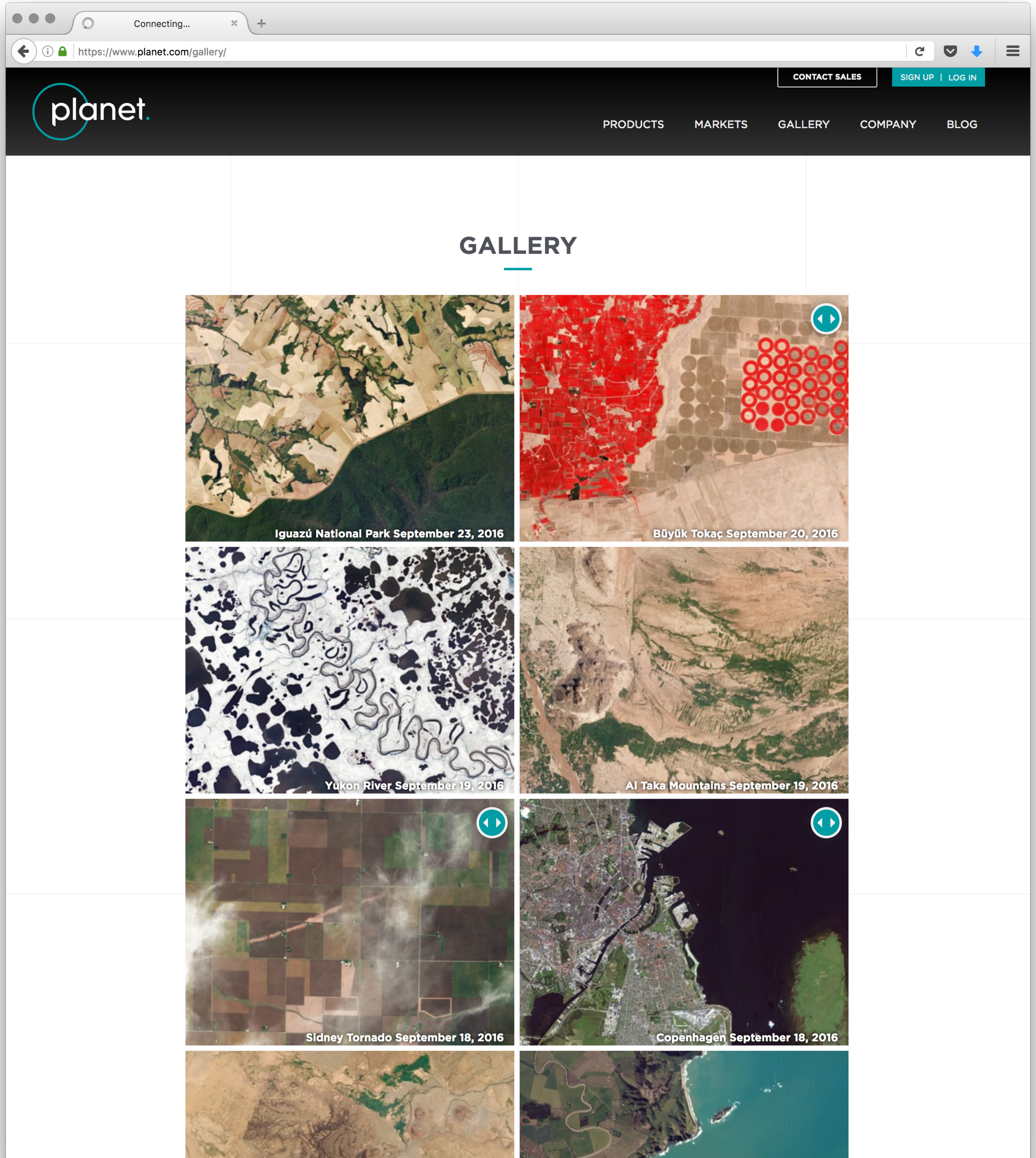
Use an **image** and control pixel colors to encode value

The value assigned for each cell represents the attribute of that cell

- **Continuous** variables, surfaces (temperature, density, elevation…)
- Satellite images (land cover, land use…)


[Source]


[Source]
CLYDE
SPACE
CS01037
CLYDE
SPACE
CS01038


Connecting...
https://www.planet.com/gallery/
CONTACT SALES
SIGN UP | LOG IN
planet.
PRODUCTS
MARKETS
GALLERY
COMPANY
BLOG
GALLERY
Iguazú National Park September 23, 2016
Büyük Tokaç September 20, 2016
Yukon River September 19, 2016
Al Taka Mountains September 19, 2016
Sidney Tornado September 18, 2016
Copenhagen September 18, 2016

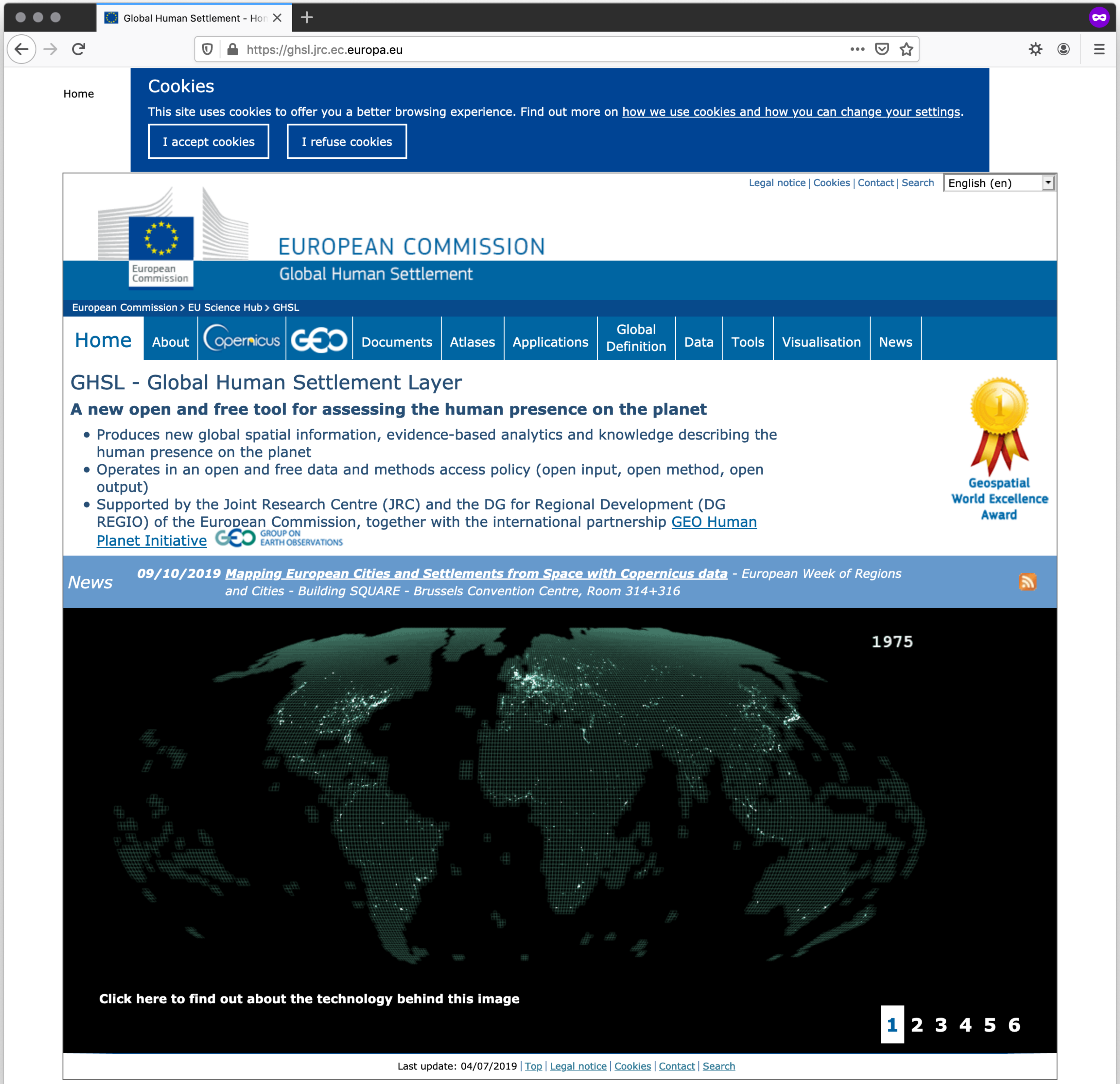
Spatial Data Formats

Spatial Data Formats

In principle…

- Points, lines, polygons –> Vector formats
- Images, surfaces –> Raster formats

But these boundaries are blur…



Global Human Settlement - Hor
https://ghsl.jrc.ec.europa.eu
Home
Cookies
This site uses cookies to offer you a better browsing experience. Find out more on how we use cookies and how you can change your settings.
I accept cookies
I refuse cookies
Legal notice | Cookies | Contact | Search
English (en)
European Commission
EUROPEAN COMMISSION
Global Human Settlement
European Commission > EU Science Hub > GHSL
Home
About
Copernicus
GEO
Documents
Atlases
Applications
Global Definition
Data
Tools
Visualisation
News
GHSL - Global Human Settlement Layer
A new open and free tool for assessing the human presence on the planet
• Produces new global spatial information, evidence-based analytics and knowledge describing the human presence on the planet
• Operates in an open and free data and methods access policy (open input, open method, open output)
• Supported by the Joint Research Centre (JRC) and the DG for Regional Development (DG REGIO) of the European Commission, together with the international partnership GEO Human Planet Initiative
GEO GROUP ON EARTH OBSERVATIONS
Geospatial World Excellence Award
News
09/10/2019 Mapping European Cities and Settlements from Space with Copernicus data - European Week of Regions and Cities - Building SQUARE - Brussels Convention Centre, Room 314+316
1975
Click here to find out about the technology behind this image
1 2 3 4 5 6
Last update: 04/07/2019 | Top | Legal notice | Cookies | Contact | Search

LUCAS - Land use and land cover survey

No update is planned for this article

The European Union (EU) is composed of a diverse range of landscapes: it is home to a wide variety of flora and fauna and includes some of the most and least densely populated areas of the world. This background article provides information on the Land Use/Cover Area frame Survey (LUCAS), a survey that provides harmonised and comparable statistics on land use and land cover across the whole of the EU's territory.

The data collected by LUCAS provides harmonised information for studying a range of socioenvironmental challenges, such as land take, soil degradation or biodiversity.

Full article +

Direct access to

Other articles | Tables | Database
Dedicated section | Publications | Methodology
Legislation | Visualisations | External links

Notes

1. ↑ https://www.sciencedirect.com/science/article/pii/S1462901115300654 The new assessment of soil loss by water erosion in Europe
2. ↑ http://onlinelibrary.wiley.com/doi/10.1111/ejss.12499/full LUCAS Soil, the largest expandable soil dataset for Europe: a review
3. ↑ https://ec.europa.eu/jrc/en/publication/lucas-2018-soil-component-sampling-instructions-surveyors LUCAS 2018 - soil component: Sampling Instructions for Surveyors

Categories: Agriculture | Agriculture background articles | Background article | Environment | Regions and cities typologies

Traditionally

`Vector`

- Single files: shapefiles, etc.
- Client-server (geo-)DBs: PostGIS, etc.

`Raster`

- Mostly (single) image formats: GeoTIFF, etc.

However…

Many of these formats were designed for an *offline* world, so display some of the following:

- Binary (“non-streamable”)
- “Unqueriable”
- Complex format structures

Shapefile must die!

switchfromshapefile.org

Fork me on GitHub

Switch from Shapefile

ESRI Shapefile is a file format for storing geospatial vector data. It has been around since the early 1990s and is still the most commonly used vector data exchange format.

While Shapefiles have enabled many successful activities over the years, they also have a number of limitations that complicate software development and reduce efficiency.

We, members of the geospatial IT industry, believe that it is time to stop using Shapefiles as the primary vector data exchange format and to replace them with a format that takes advantage of the huge advances that have been made since Shapefile was introduced.

Read more:

- The good side
- Shapefile is a bad format
- Shapefile alternatives

The good side

Shapefile does a lot of things right. Here are some reasons why Shapefile is so heavily used:

- Shapefile is by far the most widely supported format in existing software packages.
- While the format is proprietary, the specification is open.
- For many use cases, it is *good enough*.
  - Index files (e.g. *.shx) enable good reading performance.
  - It is relatively efficient in terms of file size. The resulting file, even un-zipped, is relatively small compared to some other (mostly text-based) formats.

Shapefile is a bad format

Why is Shapefile so bad? Here are several reasons why the Shapefile is a bad format and you should avoid its usage:

- No coordinate reference system definition.
- It's a multifile format.
- Attribute names are limited to 10 characters.
- Only 255 attributes. The DBF file does not allow you to store more then 255 attribute fields.
- Limited data types. Data types are limited to float, integer, date and text with a maximum 254 characters.
- Unknown character set. There is no way to specify the character set used in the database.
- It's limited to 2GB of file size. Although some tools are able to surpass this limit, they can never exceed 4GB of data.
- No topology in the data. There is no way to describe topological relations in the format.
- Single geometry type per file. There is no way to save mixed geometry features.
- More complicated data structures are impossible to save. It's a "flat table" format.
- There is no way to store 3D data with textures or appearances such as material definitions. There is also no way to store solids or parametric objects.
- Projections definition. They are incompatible or missing.
- Line and polygon geometry type, single or multipart, cannot be reliably determined at the layer level, it must be determined at the individual feature level.
- Add more ...

Modern formats

New formats have appeared in part *“fixing”* those issues, but also responding to web needs:

- Streamable (e.g. `GeoJSON`)
- Queriable (e.g. `PostGIS`/`Geopackage`)
- Single file (e.g. `.mbtiles`)

Let’s explore a bit more on a couple of them…

GeoJSON

GeoJSON

https://geojson.org

GeoJSON

GeoJSON is a format for encoding a variety of geographic data structures.

```
{
  "type": "Feature",
  "geometry": {
    "type": "Point",
    "coordinates": [125.6, 10.1]
  },
  "properties": {
    "name": "Dinagat Islands"
  }
}
```

GeoJSON supports the following geometry types: `Point`, `LineString`, `Polygon`, `MultiPoint`, `MultiLineString`, and `MultiPolygon`. Geometric objects with additional properties are `Feature` objects. Sets of features are contained by `FeatureCollection` objects.

The GeoJSON Specification (RFC 7946)

In 2015, the Internet Engineering Task Force (IETF), in conjunction with the original specification authors, formed a GeoJSON WG to standardize GeoJSON. RFC 7946 was published in August 2016 and is the new standard specification of the GeoJSON format, replacing the 2008 GeoJSON specification.

Advantages

- Plain text, (human) readable
- Streamable
- Well integrated with web standards (JSON)

Excellent exchange format

Disadvantages

- Plain text, inefficient
- Non-queriable
- Vector only

Tilesets (`.mb/vtiles`)

MBTiles | Glossary | Help | Mapl

https://docs.mapbo

Docs

Sign in

Help Glossary

Search docs.mapbox.com

MBTiles

MBTiles

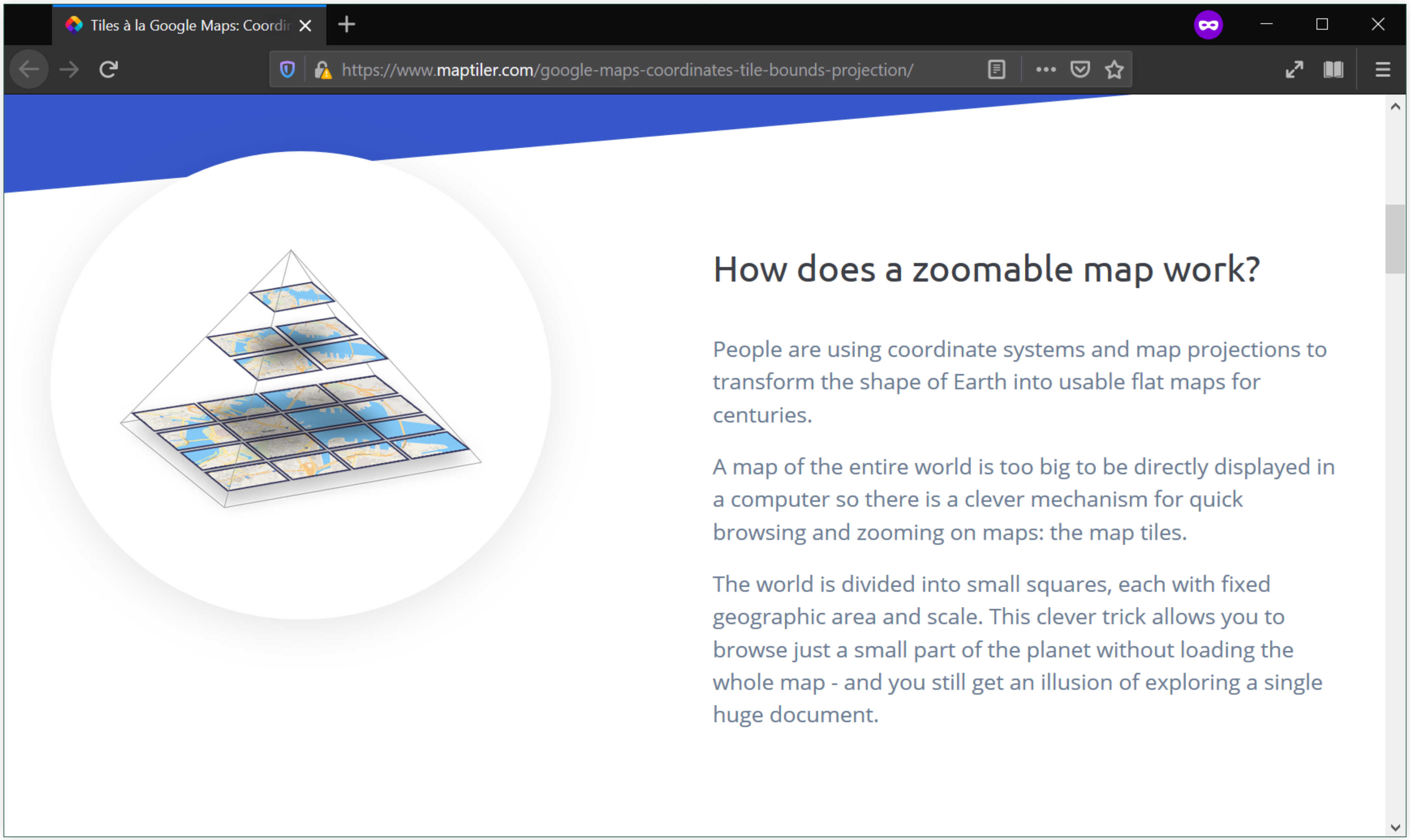
MBTiles is a file format for storing tilesets. It's designed to allow you to package up many files into a single tileset. For example, `mapbox.mapbox-streets-v8` is a single tileset that contains administrative boundaries, road networks, POIs, and other kinds of geospatial information from many different data sources.

MBTiles is an open specification based on the SQLite database. MBTiles can contain raster or vector tilesets. You can upload MBTiles files directly to Mapbox Studio or use them directly in a web or mobile application.

Related resources:

- MBTiles specification
- Studio Manual geospatial data page: transfer limits for uploading MBTiles files as tilesets

Map Tiles



How does a zoomable map work?

People are using coordinate systems and map projections to transform the shape of Earth into usable flat maps for centuries.

A map of the entire world is too big to be directly displayed in a computer so there is a clever mechanism for quick browsing and zooming on maps: the map tiles.

The world is divided into small squares, each with fixed geographic area and scale. This clever trick allows you to browse just a small part of the planet without loading the whole map - and you still get an illusion of exploring a single huge document.

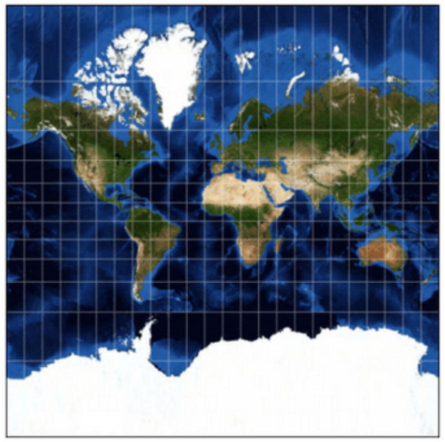
Tiles à la Google Maps: Coordi

https://www.maptiler.com/google-maps-coordinates-tile-bounds-projection/

Degrees
Geodetic coordinates
WGS84 (EPSG:4326)

Longitude and latitude coordinates used by GPS devices for defining position on Earth using World Geodetic System defined in 1984 (WGS84).

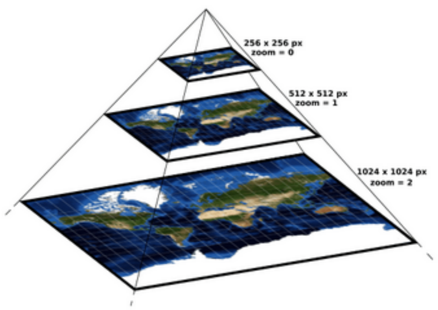
HINT: WGS84 geodetic datum specify lon/lat (lambda/phi) coordinates on defined ellipsoid shape with defined origin ([0,0] on a prime meridian).



Meters
Projected coordinates
Spherical Mercator
(EPSG:3857)

Global projected coordinates in meters for entire planet. Used for raster tile generation in GIS and WM(T)S services.

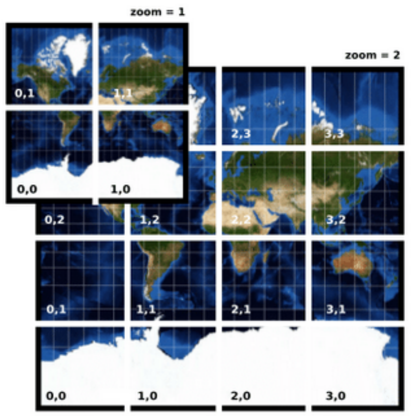
HINT: Simpler spherical calculation are used instead of ellipsoidal. Mercator map projection deforms size (Greenland vs Africa) and never shows poles.



Pixels
Screen coordinates
XY pixels at zoom

Zoom-specific **pixel coordinates** for each level of the pyramid. Top level (zoom=0) has usually 256x256 pixels, next level 512x512, etc.

Devices calculate pixel coordinates at defined zoom level and determine visible viewport for area which should be loaded from servers.



Tiles
Tile coordinates
Tile Map Service (ZXY)

Coordinates of a **tile in the pyramid.** There is one tile on the top of the pyramid, than 4 tiles, 16 tiles, etc. All tiles have the same size, usually 256x256 pixels.

Only the relevant tiles loaded and displayed for the area of interest / viewport.

Advantages

- Queriable (SQLite)
- Fast access to large maps with limited resources (client/server model + queriable format)
- Some (vector tiles) are stylable

Disadvantages

- Designed for *serving* not *analysing*
- A dataset needs to be stored at several zoom levels
- Once created, hard to modify (e.g. reproject)

Selecting the right format

Selecting the right format

No silver bullet…

- *What* type of data do you want to store? Vector, raster
- *What* are you going to do with the file? Analysis, serving
- *What* environment are you working? Locally, web

Quiz

1. Large dataset of tweets you want to analyse `PostGIS/Geopackage`
2. Drone imagery to make available for workshop participants `MBTiles`
3. Street basemap to provide context to a small dataset you want to make available on the web `Vector tiles`
4. The small dataset from 3. `GeoJSON`